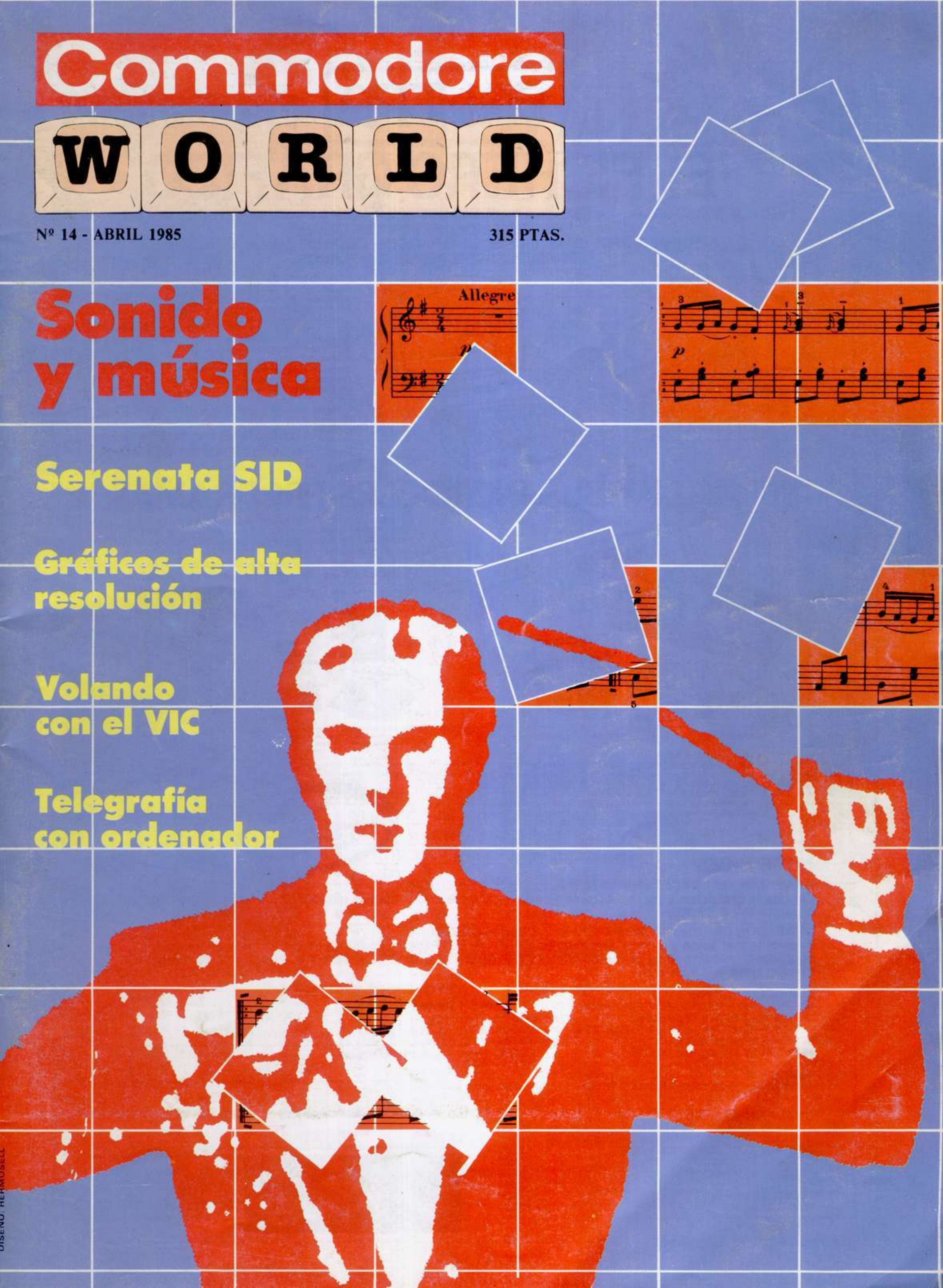
Commodore
WORLD
Nº 14 - ABRIL 1985
315 PTAS.
Sonido y música
Serenata SID
Gráficos de alta resolución
Volando con el VIC
Telegrafía con ordenador
Allegro
DISEÑO: HERMOSELL

SUPER-INTERESANTISIMO

PIRATERIA DE SOFT

—Comienza la polémica—

Con el número en la imprenta y ya cerrado, hemos recibido dos cartas, una de A.J.O.G. (pide no se publiquen sus datos) de San Sebastián y de José Emilio Bases Muñoz de Zaragoza sobre el editorial publicado en nuestro número 13 del mes de marzo, dedicado a la piratería de Soft.

Consideramos que estas cartas son de suficiente interés para publicarlas en su totalidad y contestarlas debidamente, por lo que les dedicaremos el espacio necesario en nuestro próximo número de mayo.

Commodore World en discos

¡Ya! Desde este número 14 del mes de abril, como ya anunciamos en febrero, a partir de hoy editamos la revista en disco, paralelamente a la edición impresa. Recordamos que el disco irá en su estuche con su portada correspondiente a todo color.

El precio de cada disco es de 2.000 ptas. y 1.750 ptas. para los suscriptores.

La suscripción anual de 11 números es de 1.750 ptas. La forma de pago es *solamente* por cheque o giro, y encartado en la revista tenéis el boletín de pedido.

Tapas autoencuadernables

Las tapas —Volumen 1— para doce números de Commodore World estarán ya disponibles a partir del 20 de abril. Estas tapas son integradas y no necesitan encuadernación posterior, ya que interiormente llevan unas pestañas para insertar directamente las revistas por su página central. Su precio es de 450 ptas., la forma de pago es, *solamente*, por cheque o giro y encartado en la revista tenéis el boletín de pedido.

Números agotados atrasados

1, 3, 4, 5, 6 y especial 100 programas.

RAMY Y ROMO

Como veréis, en este número, por razones de espacio nos vemos obligados a quitar el comic que habitualmente aparece en nuestras páginas dirigidos a los peques —de dos a noventa años—. Esperamos que en el próximo número podamos disfrutar de la compañía de nuestros amigos.

HACIA EL SUSCRIPTOR 8.000
RECORDATORIO

Sorteo entre TODOS los suscriptores de:

1) Viaje para dos personas, a elegir entre París, Lisboa, Atenas, Amsterdam, Niza, Viena, Canarias, Mallorca.

2) Impresora donada por Microelectrónica y Control.

3) 80 programas valorados en 160.000 ptas. donados por la casa POKE, S.A. de Barcelona.

ULTIMO NUMERO
DE SUSCRIPTOR = 7.001

CONCURSOS PERMANENTES

Colaboraciones de Programas y Magia

La verdad sea dicha: os estáis portando; las colaboraciones que nos están llegando son numerosas y de muy buena calidad. Por lo tanto, vamos a aclarar las condiciones de nuestros concursos permanentes.

1) Los sorteos se realizarán en los meses de junio y diciembre.

2) Se sortearán siempre ocho premios, cuatro de 15.000, 10.000, 5.000 y 3.000 pesetas en metálico de "Commodore World" y cuatro del mismo valor en material de Microelectrónica y Control, entre todas las colaboraciones publicadas.

3) Se sortearán siempre seis premios, tres de 3.000, 2.000 y 1.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las contribuciones publicadas en la sección de "Magia".

4) Adicionalmente, se sortearán diversos premios de cintas, juegos, objetos curiosos, etc., entre todas las colaboraciones.

5) Los autores de las colaboraciones vendidas dentro de nuestro "Servicio de Cintas" percibirán el 20% del precio de la cinta.

6) Todas las colaboraciones deben venir escritas a máquina y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestros lectores más jóvenes pueden escribir a mano pero con letra muy clara.

7) Quedan automáticamente descalificadas tanto del sorteo como del "Servicio de Cintas" las colaboraciones que hayan sido enviadas a otras revistas.

8) Las colaboraciones se enviarán a "Commodore World", c/Barquillo, 21-3º izda. 28004 Madrid.

METEDURAS DE PATA

- En la sección de Magia, de nuestro número 13, en el programa titulado "números encolumnados", aparecía dos veces un cuadradito inverso entre comillas. Habréis adivinado que era un "[CLR]".
- El número de teléfono de IDEALOGIC apareció equivocado en una de las reseñas, el número correcto es el (93) 330-33-08. Mil perdones al propietario del teléfono equivocado.
- En el listado del programa "reductor", las últimas líneas (de la 60000 en adelante) no tienen nada que ver con el programa, es una rutina que utilizamos nosotros para grabar las versiones de un programa en disco. Si alguno quiere utilizarla, sólo tiene que colocar el nombre del programa en la línea 60005 y hacer RUN60000 cada vez que quiera salvarlo. En el disco hay siempre dos versiones del programa, la ".bkup" es la más antigua y la ".last" la última. "No hay mal que por bien no venga".

Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas que aparecen del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD]= Tecla cursor hacia abajo (sin SHIFT)
[CRSRU]= Tecla cursor hacia arriba (con SHIFT)
[CRSRR]= Tecla cursor a la derecha (sin SHIFT)
[CRSRL]= Tecla cursor a la izquierda (con SHIFT)
[HOME]= Tecla CLR/HOME (sin SHIFT)
[CLR]= Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora. •

INDICE DE ANUNCIANTES

Contabilidad - 64 así de fácil.

El programa Contabilidad 64 es líder en ventas, por su sencillez, rapidez, eficacia y precio.

COMO INICIAR LA ENTRADA DE DATOS EN EL PLAN CONTABLE



Seleccione el número 6 de «menú general» y aparecerá en pantalla el submenú «utilitarios». Llame el número 1 y haga una copia del disco. El original no se debe utilizar como disco de trabajo.

Contabilidad 64 se ha realizado según el Plan General Contable Español y ha sido desarrollado especialmente para los microordenadores Commodore 64 y EXECUTIVE.

El programa tiene capacidad para 300 cuentas y un número ilimitado de apuntes o registros por cuanto el programa permite generar nuevos discos en los que continuar el ejercicio contable.

Contabilidad 64 permite regenerar los índices de todos los ficheros, verificar el disco, etc. Estas utilidades hacen de Contabilidad 64 un producto de software autosuficiente. Asimismo Contabilidad 64 permite en todo momento tener acceso a los ficheros de manera que puedan modificarse los datos contenidos en ellos, aún cuando estos datos ya hayan sido validados; esta posibilidad da una gran libertad al usuario en el manejo de la información.

Por todo ello se lo damos a conocer paso a paso.

1. Ficheros que componen la contabilidad y su estructura.
2. Cómo se estructura el programa Contabilidad 64.
3. **Cómo iniciar la entrada de datos en el Plan Contable.**
4. Métodos para la introducción de asientos contables y obtención del Diario.
5. Balances de sumas y saldos y de situación.
6. Cómo consultar las cuentas y obtención del Mayor.
7. Cuentas de explotación y Diario de cierre.
8. Mantenimiento de ficheros, listados y otras utilidades.

Seguidamente seleccione el número 4 «mantenimiento de ficheros» y llame la opción 1 «parámetros», aparecerá «mantenimiento de ficheros» con el cursor intermitente en el lugar donde se debe colocar el nombre de titular de la contabilidad. Mediante la tecla «return» descenderá de línea en línea y automáticamente se irán actualizando los campos.

Para dar de alta las cuentas (tres dígitos) y subcuentas (seis dígitos) seleccione la opción 2 «cuentas».

Si precisa conocer todas las ventajas hoy, envíenos el cupón adjunto y recibirá gratuitamente el manual resumido de Contabilidad 64.

casa de software s.a.

c/. aragón, 272, 8.°, 6.ª
tel. 215 69 52
08007 barcelona

DESEO RECIBIR INFORMACION:

Nombre y dirección:

____

____

____

Commodore 64 es una marca registrada de Commodore Bussines Machines, Inc.

SONIDO

Casi todos los ordenadores son capaces de producir algún tipo de sonido, pero, a diferencia de los otros, el C-64 dispone de un sonido que puede ser manipulado, cambiado y controlado desde el teclado —no necesita más hardware—.

La flexibilidad y el bajo precio del C-64 no son las únicas cosas que le convierten en una máquina de hacer música de alta calidad. El secreto es un pequeño trozo de arena preparada llamado el SID (Sound Interface Device) Interface de Sonido.

El SID es uno de los más sofisticados sintetizadores de música disponibles en cualquier ordenador personal. Dispone de tres voces, cada una con su envolvente correspondiente de ataque/decaimiento; sostenimiento/relajación (ADSR); filtros, modulación de timbre y sincronización; y cuatro generadores de forma de onda. Pero esto no es todo.

Casi todos los ordenadores son capaces de producir algún tipo de sonido, pero, a diferencia de otros generadores, todas las características del SID pueden ser manipuladas, modificadas y controladas desde el teclado, sin la necesidad de usar más hardware. Como por arte de magia, el SID produce prácticamente cualquier cosa que tú puedas imaginar.

Para poder aprovechar al máximo este sintetizador en un chip, tienes que saber algo acerca de la música y la programación en Basic. Las cosas se ponen más difíciles si no sabes nada de la una y menos todavía de la otra. Sigue leyendo —aunque ritmo te suena a coche y bucle a peinado— estamos aquí para ayudarte.

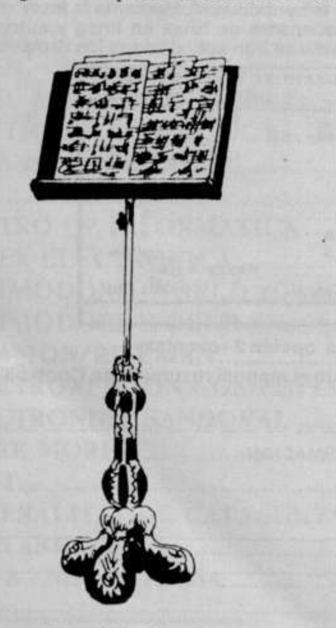
Aprendes a tocar el sintetizador de la misma forma en que aprendes a tocar cualquier otro instrumento musical: prácticas.

Teclea NEW, pulsa la tecla Return y teclea el Listado 1. Sálvalo en disco o cinta antes de ejecutarlo.

Se utiliza la sentencia Poke con frecuencia. Antes de seguir leyendo este artículo, te aconsejo que leas el apartado sobre Pokes y Peeks.

Descripción del Programa

El programa empieza en la línea 10 con un REM. En la línea 20, se establece que S representa 54272 durante el resto del programa. A continuación se apaga el chip SID introduciendo ceros mediante unos Pokes en todos los registros que contengan un bucle For...Next. Durante la primera pasada del bucle, I es igual a 0 (FOR I=0 TO 28); se añade a S (S+I), y se introduce mediante un Poke en 54272 (POKE S+I,0). Cuando el programa encuentra la sentencia Next, vuelve automáticamente a la última sentencia For y aumenta el valor de I por 1; a continuación 0 se introduce en 54273 (S+1) mediante un Poke. Cuando se vuelve a encontrar la sentencia Next, I se convierte en 2, un cero se introduce en 54274 mediante un Poke, etc., hasta que el chip entero quede borrado o apagado.



La línea 30 usa un Poke para introducir la envolvente ataque/decaimiento y sostenimiento/relajación en los registros correctos para la voz 1. Puedes modificar el tipo de sonido cambiando los números que siguen S+5 o S+6. La A/D y S/R, utilizada aquí junto con la onda de diente de sierra, producen un sonido parecido a un banjo.

En la línea 40 el volumen interno sube hasta 15, su punto máximo. Ajusta el volumen en tu televisor o monitor hasta encontrar un nivel cómodo.

Sentencias Read y Data

Cuando el C-64 encuentra una sentencia Read (línea 50), busca automáticamente una sentencia Data que le acompaña (que puede estar en cualquier lugar del programa, en este caso, las líneas 200-300). Las sentencias Read y Data trabajan en equipo: una no puede existir sin la otra. Observa que la sentencia Read va seguida de dos nombres de variables: HF, que contiene 16, y LF, que contiene 195. Imagínate que son etiquetas en las cajas que contienen los datos.

Cuando se encuentra la sentencia Read, seguida de dos variables, el ordenador mira la primera sentencia DATA que encuentre en el programa y recogera las dos primeras unidades después de la palabra DATA. El ordenador sabe que los datos van separados por comas.

De momento sáltate la línea 60. La línea 70 usa un Poke para introducir los datos contenidos en HF y LF en los registros de alta y baja frecuencia para la voz 1. Observa que READ HF se encuentra primero (línea 50), pero el ordenador va primero a POKE LF (línea 70). Yo podía haber invertido el orden de esto, pero quiero que veas que algo parece ocurrir al reves en un ordenador. Lo importante es usar un poke para introducir el valor de HF en el registro de alta frecuencia y el valor de LF en el registro de baja frecuencia. Si se hace al revés, se producen chirridos en vez de tonos cuando ejecutas el programa.

¿Qué ha pasado hasta ahora? Se ha limpiado el chip SID, fijado la envolvente ADSR, encendido el volumen e introducido los números que controlan la nota. Pero no suena nada todavía.

El sonido se produce a partir de la línea 80. En esta línea se selecciona y se enciende la forma de onda, y hasta que no se haga esto, no se produce ningún sonido.

Cuando se ejecuta la línea 80, suena una nota. Un bucle de retraso en la línea 90 hace que la nota no suene para siempre. Le indica al ordenador que cuente 200 "jiffies" antes de ejecutar la siguiente instrucción. Un "jiffie" es un 1/60 de segundo en tiempo real, de modo que 200 "jiffies" no es mucho tiempo. Después de contar hasta 200, el programa va a la línea 100, donde el sonido se apaga introduciendo 32 en S+4 mediante un Poke, lo que quita el 1 que encendió el sonido en la línea 80. Se sigue generando la onda de diente de sierra, pero ahora no se oye. Después de apagar el sonido, el ordenador cuenta 10 "jiffies" (FOR I=1 TO 10:NEXT) antes de pasar a la siguiente instrucción. Esto intercala un silencio muy breve entre los tonos.

El Bucle GOTO

El principio de un bucle GOTO se encuentra en la línea 110. Mientras que los números de línea establecen el orden en que el ordenador ejecuta las instrucciones, un GOTO interrumpe dicho orden y envía el programa a una línea que queda fuera de secuencia. GOTO 50 significa volver a la línea 50 y ejecutar las instrucciones que se encuentran allí. De esta forma, se vuelve a iniciar el proceso Read/Data, pero esta vez con la segunda unidad de datos, ya que esta es la segunda vez que se ha ejecutado la instrucción Read en la línea 50. El programa sigue

C-64
VIC-20

Y MUSICA

ejecutando este bucle GOTO hasta que termine con todos los datos, pero daría un Syntax Error si no estuviera la línea 60 (la que saltaste antes).

El poder de IF...Then

Un ordenador programado en Basic puede tomar decisiones gracias a la sentencia If...Then. Funciona de la siguiente forma: Si (IF) algo es verdad, entonces (Then) hay que hacer una cosa determinada. Cuando la condición comprobada no es correcta (cuando HF no es igual a —1 en este programa), se ignora la instrucción que sigue THEN (junto con el resto de la línea). La línea 60 de este programa comprueba si el valor de HF es igual a —1, que se utiliza como una bandera. Ahora acuérdate, cada vez que se leen nuevos datos, y el programa vuelve a la línea 50, se modificará el valor en las cajas etiquetadas HF y LF. Y cada vez, el valor de HF será comparado con —1 gracias a la línea 60. Si miras la línea 200, verás que los dos últimos datos son —1, —1. Se necesitan dos unos (1s) negativos porque el ordenador recoge dos grupos de datos a la vez (READ HF,LF), y el programa dejara de ejecutarse y te enviará un mensaje de error si no puede recoger los dos. Cuando lee este último par de datos, volvera a comprobar si HF es igual a —1, y esta vez será cierto. A continuación, y siguiendo la instrucción después de THEN en la línea 60, se apagará el volumen y se finalizara el programa. Esta es la forma de salir de un bucle GOTO.

Jugando un poco
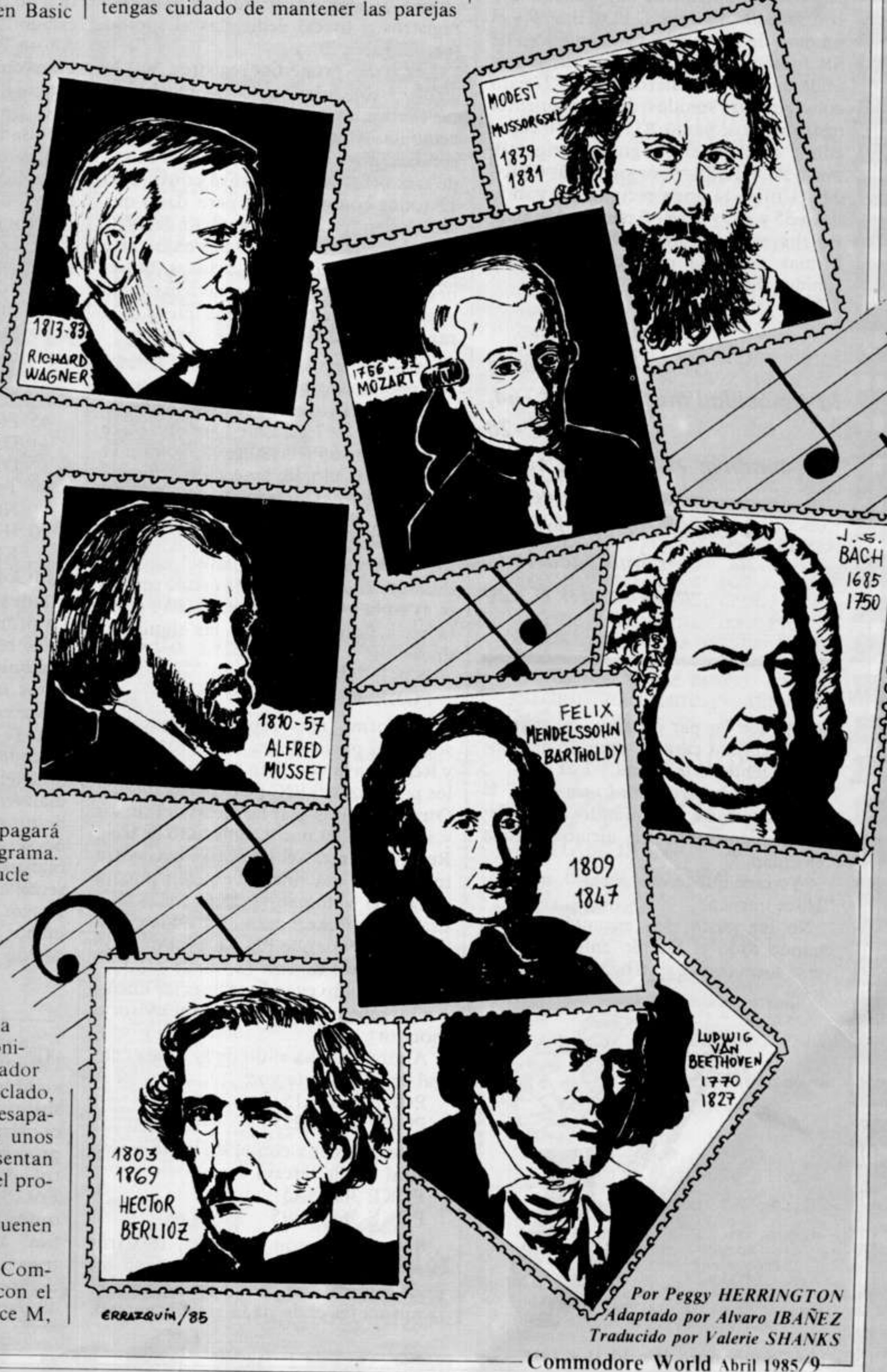
Cuando tienes el programa funcionando, escúchalo y experimenta un poco, produciendo diferentes sonidos. No puedes dañar el ordenador introduciendo datos a través del teclado, y si el ordenador queda colgado (desaparece el cursor), apágalo durante unos segundos. A continuación se presentan algunos métodos para modificar el programa:

1) Modificar los datos para que suenen notas diferentes.

Mira "La Guía del Usuario de Commodore" (la que recibiste junto con el ordenador) y consulta al Apendice M, Valores de las Notas Musicales, en la página 152. Tal y como está el programa ahora, suena una octava más alto, empezando con C-4 (el do de la escala central). Los datos para el programa se encuentran en la página 153. Busca C-4 en la columna Nota-Octava y sigue hacia la derecha en las columnas de Alta Frecuencia y Baja Frecuencia, donde encontrarás los dos primeros datos de la línea 200. Puedes cambiar los datos (y hasta añadir más líneas al programa) con tal de que tengas cuidado de mantener las parejas de HF y LF y que los dos últimos datos sean —1.

2) Cambiar la forma de onda

Este programa utiliza una onda de diente de sierra (valor 32), que se activa en la línea 80 y se desactiva en la línea 100. Si cambias la forma de onda, se cambiará la calidad del sonido. Prueba con la forma de onda triangular (valor de 17 en la línea 80 y de 16 en la línea



Por Peggy HERRINGTON
Adaptado por Alvaro IBAÑEZ
Traducido por Valerie SHANKS

100) o el ruido (129 en la línea 80, 128 en la línea 100). Puedes usar la onda de pulso también (valor de 65 en la línea 80 y de 64 en la línea 100), pero añade esta línea al programa:

35 POKE S+3,10:POKE S+4,150:REM ANCHURA DE PULSO

3) Cambiar el generador de envoltura ADSR

Se pueden imitar diferentes instrumentos musicales si tienes los valores ADSR o simplemente experimentando con sonidos diferentes. POKE S+5 y el número de AD y S+6 con el número de SR (ambos en la línea 30), utilizan la onda de diente de sierra (valor 32) para conseguir los sonidos de los siguientes instrumentos: banjo AD=3,SR=0 (tal y como está en el programa original), piano AD=10,SR=9, órgano AD=0,SR=240. Utiliza la onda rectangular con la línea 35 y AD=9,SR=0 para la guitarra. Prueba estos valores ADSR con otras formas de onda para conseguir más sonidos distintos.

Se habla mucho de la capacidad musical del C-64, pero el VIC-20 también es capaz de tocar un par de cosas. A continuación veremos cómo sacarle la música al VIC-20

Hace un par de años, mi marido trajo a casa un aparato aparentemente inocente.

"¿Qué te has comprado un qué?" le dije. Lo estaba enchufando a la televisión mientras yo me alejaba a toda velocidad.

"Yo creo que te va a gustar", me dijo. "Hace música".

No me gustó. Me encantó. Incluso cuando todo el mundo empezó a volverse loco con el sintetizador del C-64, yo me mantuve firme en defensa de mi pequeño VIC.



Para que el VIC-20 produzca sonidos, tienes que saber tanto de música como de programación Basic.

El VIC dispone de cuatro voces —tres de ellas son musicales y una es ruido blanco (lo que se oye entre canal y canal en una televisión). No hay muchos controles de sonido complicados, y dispone de todo lo necesario para hacer música y efectos de sonido. Existen cinco posiciones de memoria (también llamadas registros o bytes) dedicadas al sonido (ver la Tabla 2).

Los tres primeros registros en la Tabla 2 controlan las notas musicales, los tonos que componen la melodía y la armonía. A propósito, C4 es el do de la escala central. Cada voz tiene un rango de tres octavas (una octava consiste en 12 tonos consecutivos), pero dado que se solapan, el rango completo del VIC —es decir, desde el tono más bajo hasta el más alto— son cinco octavas. Si tienes un piano, ignora la octava más baja y las dos más altas, y lo que queda es el rango del VIC.

Para que el VIC produzca sonido, hay que activar el volumen interno del ordenador. Esto se hace introduciendo un número de 1 a 15 mediante un poke en el registro 36878. (Antes de continuar con este artículo, lee la sección dedicada a pokes y peeks). Después de activar el volumen, el número correspondiente a la nota elegida se introduce en uno de los registros de voz mediante un poke. Para que toque el do de la escala central (C4) en la voz baja, teclea NEW, pulsa la tecla Return y teclea las siguientes líneas:

```
POKE 36878,15
POKE 36874,240
```

La forma más rápida de pasar el sonido es pulsando las teclas Run/Stop y Restore a la vez. Esto hace un Reset en los punteros del VIC y borra la pantalla. Otro método es usar las teclas del cursor y sustituir 240 por 0, pulsando la tecla Return mientras el cursor se encuentra todavía en esa línea. También puedes apagar el volumen (POKE 36878,0) o usar un Poke para introducir un número menor que 128 o mayor que 255 en el registro de voz. (Si no se produce ningún sonido cuando tecleas las líneas, comprueba el volumen de tu televisor o monitor).

Ahora escucha el do de la escala central en la segunda voz:

```
POKE 36878,15
POKE 36875,225
```

Ahora prueba con el do de la escala central con la tercera voz:

```
POKE 36878,15
POKE 36876,195
```

Si escuchas con atención, te darás cuenta de que estos tonos suenan un poco diferente en cada voz. La nota es la misma (es el do de la escala central), aunque los números 240, 225 y 195 son distintos (porque los rangos de las voces se solapan), pero la calidad del sonido, lo que los músicos llaman color del tono o timbre, no es exactamente la misma. Las voces 1 y 2 son más suaves que la tercera. Es así para que la melodía, que normalmente se toca en tonos más altos cuando se sincroniza con otras voces, se destaque sobre ellas.

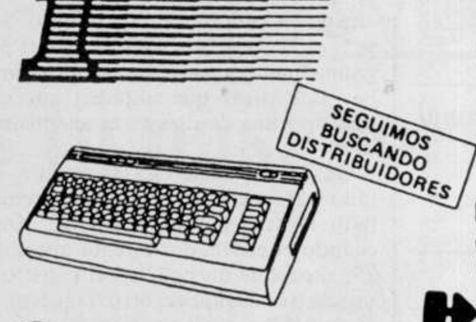
El VIC-20 es capaz de producir muchas notas que un instrumento acústico no puede. Hasta ahora el VIC te ha estado cantando en modo directo. Ahora vamos a cambiar al modo programa para escuchar todas las notas (el rango) que tiene el registro más bajo del VIC. Borra la memoria tecleando NEW y pulsando Return, y teclea el siguiente programa. Teclea incluso los REM (el ordenador ignora todo lo que vaya detrás de un REM en una línea) hasta que comprendas lo que está ocurriendo en el siguiente programa:

```
10 S=36874:V=36878:T=128
20 POKE V,10:REM ENCENDER
   VOLUMEN
30 POKE S,T:REM SONIDO
40 FOR D=1TO100:NEXT D:REM
   BUCLE DE DURACION DEL
   SONIDO
45 POKE S,0:FOR T=1TO50:NEXT
   D:REM BUCLE DE DURACION
   DEL SILENCIO
50 T=T+1:REM  INCREMENTAR
   NOTA
60 IF T >253 THEN POKE S,0:END:
   REM COMPROBAR FIN
70 GOTO 30
```

Salva el programa antes de ejecutarlo. La primera línea inicializa las variables y las constantes. Podías haber elegido cualquier letra (o combinación de dos letras o una letra y un número) para representar la voz (S), volumen (V) o el tono (T), pero estas letras te ayudarán a recordar lo que va en el programa más tarde. El uso de las variables, que normalmente cambian de valor a lo largo de un programa, y las constantes, que no se cambian, en vez de números de registro, tiene varias ventajas. Son más sencillas de teclear, lo que reduce los riesgos de cometer un error, utilizan menos memoria y el VIC trabaja más rápido con ellas que con números.

Bucles For...Next

Se enciende el volumen en la línea 20. La línea 30 produce el primer sonido audible mediante un poke (el ordenador sustituye automáticamente el valor numérico actual por una variable en un programa, en este caso 128) en el registro de la voz 1 (36874). El bucle For...Next en la línea 40 le indica al ordenador que cuente hasta 100 "jiffies" antes de ejecutar la siguiente instrucción. De esta forma se controla el tiempo que dura el sonido (quita la línea 40 para ver qué ocurre). ▶



FIRST S.A.

C/ Aribau, 62. BARCELONA- 08011
Tel. (93) 323 03 90
Tlx. 53947 FIRS E (ESPAÑA)

GRAN ¡NOVEDAD!

COMPUPRO

FLOPPY DISK DRIVE PARA COMMODORE 64 COMPATIBLE

54.000 ptas.



COMMODORE 64 + ... = PTAS.

ErgoVisión (Base móvil)

26.000 PTAS

The Monitor

19.000 pts.

MICRO AGE

DD/SC 3.995 PTAS.
DD/DC 4.550 PTAS.

FIRST S.A. Importador para España de MICRO AGE

para **COMMODORE 64**

LA POTENCIA DE UN PAQUETE GRAFICO

KoalaPad Touch Tablet

22.105 PTAS.

FIRST IMPRESORAS MAS PERSONALES star

NUEVO SG-10

79.500 pts.

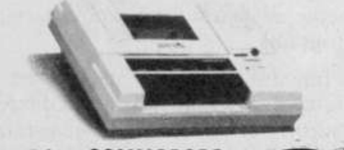
GEMINI 15 X 115.000 Pt

INCLUYE LA INTERFACE MICROANGELO Y CONECTORES

Y AHORA...
TAMBIEN MICROANGELO
PARA COMMODORE 64
CON GRAFICOS Y SIGNOS

EXTRA COMPUTER

DATA RECORDER



PARA COMMODORE VIC 20 64

9.500 pts.

SUPERJOYSTICK +



3.500 Ptas.

SUPER JOYSTICK

COMPATIBLE CON
COMMODORE 64 y VIC 20
SINCLAIR (precisa interf)
ATARI
SPECTRAVIDEO

2.200!!

LOS LIBROS DE 1



LIBROS PARA COMMODORE:

CURSO DE INTRODUCCION AL BASIC PARTE I	1.950 ₧
CURSO DE INTRODUCCION AL BASIC PARTE II	3.000 ₧
GUIA DE REFERENCIA DEL PROGRAMADOR	1.950 ₧
COMMODORE 64 GUIA DEL USUARIO	2.100 ₧
BASIC PARA COMMODORE 64. MANUAL PRACTICO	1.620 ₧
COMMODORE 64. PASATIEMPOS Y JUEGOS	1.290 ₧
COMMODORE 64 QUE ES, PARA QUE SIRVE Y COMO SE USA	950 ₧
18 JUEGOS DINAMICOS PARA TU COMMODORE 64	650 ₧
64 COMMODORE APLICACIONES PARA LA CASA Y PEQ. NEGOCIOS	830 ₧
COMMODORE 64 SELECCION DE JUEGOS	950 ₧
VIC 20 GUIA DEL USUARIO	2.300 ₧
ACCESO RAPIDO AL VIC 20	1.200 ₧
LOS LIBROS RECOMENDADOS POR FIRST EN INGLES PARA COM64:	
MAPPING THE COMMODORE 64	3.095 ₧
MACHINE LANGUAJE ROUTINES FOR THE COMMODORE 64	3.095 ₧
TOOL KIT: BASIC for THE VIC 20 and COMMODORE 64	3.510 ₧
ALL ABOUT THE COMMODORE 64: VOL. I	2.680 ₧
COMMODORE COLLECTION: VOL I.20 ORIGINAL PROGRAMS	2.680 ₧
BEGINNERS GUIDE TO COMMODORE 64 SOUND	2.680 ₧
FIRST BOOK OF COMMODORE 64 (Applications, Utilities, etc.)	2.680 ₧
REFERENCE GUIDE TO COMMODORE 64 GRAPHICS	2.680 ₧
FIRST BOOK OF 64 SOUND AND GRAPHICS	2.680 ₧
FIRST BOOK OF COMMODORE 64 GAMES	2.680 ₧
COMMODORE 64 GAMES FOR KIDS	2.680 ₧
COMMODORE 64 BASIC PROGRAMS. Le mostrará como trabajar con su 64. Incluye cassete y disco con programas	4.290 ₧
DATA HANDLING ON THE COMMODORE 64 MADE EASY. Pantallas, menus, archivos, busquedas de datos en su 64, resuelto	1.870 ₧
COMMODORE 64 BASIC PROGRAMS	4.290 ₧
COMMODORE 64 DISK COMPANION.ESSENTIAL ROUTINES	2.500 ₧
SOFTWARE FOR THE COMMODORE 64. BEST OF PCW	3.760 ₧
OTROS LIBROS DE FIRST:	
APRENDIENDO CON LOGO	1.490 ₧
VIDEO JUEGOS (Catalogo de los Juegos para ordenador)	750 ₧
SISTEMA OPERATIVO CP/M. GUIA DEL USUARIO. 2/ ED	2.280 ₧
DISCO GUIA CP/M	650 ₧
PASCAL ESTRUCTURADO	2.350 ₧
INTRODUCCION AL UCSD PASCAL	1.520 ₧
PROGRAMACION EN PASCAL	1.780 ₧
ALGUNOS PROGRAMAS DE USO COMUN EN PASCAL	1.950 ₧
PROGRAMAS PRACTICOS EN PASCAL	1.900 ₧
BASIC BASICO	1.300 ₧
BASIC BASICO. CURSO DE PROGRAMACION. 9a.Edic. Castellano	1.000 ₧
MANUAL DE BASIC	350 ₧
POGRAMACION BASIC.(Gottfriedo)	1.450 ₧
POGRAMACION BASIC PARA MICROCOMPUTADORAS (Joyanes)	1.490 ₧
BASIC AVANZADO PARA NIÑOS. CON NOTAS DIDACTICAS	625 ₧
ALGUNOS PROGRAMAS DE USO COMUN EN BASIC	1.900 ₧
PROGRAMAS PRACTICOS EN BASIC	1.900 ₧
SISTEMAS ADMINISTRACION DE BASE DE DATOS	1.850 ₧
ESTRUCTURA DE DATOS. PROGRAMACION Y APLICACIONES	2.300 ₧
GLOSARIO DE COMPUTACION	2.100 ₧

PAPEL ESPECIAL PARA VD.

2.500 Hojas de papel (bordes perfectos). Superblanco. Grueso. 80c	4.456 Ptas
2.500 Recibos negociables standard	4.663 Ptas
1.000 Etiquetas autoadhesivas 2 x linea)	828 Ptas

Todos los envios de papel son a portes debidos



Paddle Pair

COMMODORE

1.900 Pts.

Ordenadores más Personales!



METHAMORPHIC

RAM: 48 Kb, AMPLIABLE HASTA 192 KB.
ROM: 12 KB.
TECLADO: ASCII, TIPO MAQUINA DE ESCRIBIR CON TECLADO NUMERICO CON SIGNOS MATEMATICOS (IDEAL PARA SUS HOJAS DE CALCULO ELECTRONICAS).
MAYUSCULAS Y MINUSCULAS.
40 x 24 CARACTERES POR PANTALLA (OPCIONAL 80 x 24, 128 x 32, 132 x 24 Y 160 x 24).
GRAFICOS: MEDIA RESOLUCION Y ALTA RESOLUCION (280 x 192 PUNTOS).
COMPATIBLE CON LOS PROGRAMAS DEL MUNDO DE

EXPANSION: POR MEDIO DE 8 CONECTORES, SE ABRE AL MUNDO DE: DISK DRIVES, DISCOS WINCHESTER, PAL COLOR, 80 COLUMNAS, Z-80, PLOTTER, IMPRESORAS (MARGARITA Y MATRICIAL), KOALA PAD, ETC.
SISTEMAS OPERATIVOS: DOS 3.2, DOS 3.3, PRODOS, PASCAL, CP/M.
INCLUYE : MANUAL COMPLETO EN CASTELLANO Y EN LA VERSION DISCO: WORLD PACK (R) (TRATAMIENTO DE TEXTOS, BASE DE DATOS, HOJA DE CALCULO ELECTRONICA, TUTOR DE BASIC, EDITOR DE PROGRAMAS Y UTILIDADES, PAQUETE DE JUEGOS).

METHAMORPHIC

79.500 pts.

METHAMORPHIC +
Disco COMPUPRO +
con Controlador
Monitor 12" Fosforo Verde

147.000 pts.

DISTRIBUIDOR GENERAL EXCLUSIVO PARA ESPAÑA DE

METHAMORPHIC



ENTREGA INMEDIATA A PROVINCIAS

SOLICITE NUESTRO CATALOGO
× CATALOGO DE SOFT, HARD Y LIBROS
× TRES CATALOGOS EN UNO
VEA NORMAS EN VENTA POR CORREO

VENTA POR CORREO

MANDE SU PEDIDO. PAGO TALON CONFIRMADO O GIRO POSTAL. PEDIDOS INFERIORES 4.500 ₧ SUME 180 ₧ GASTOS ENVIO. LIBROS NO SUME GASTOS. CATALOGO COM64 MANDE 200 ₧ GASTOS ENVIO. LA PRESENTE LISTA ES SUSCEPTIBLE DE SER MODIFICADA SIN AVISO PREVIO

El VIC dispone de cuatro voces —tres de ellas son musicales y una es ruido blanco (lo que se oye entre canal y canal en una televisión). No hay muchos controles de sonido complicados, y dispone de todo lo necesario para hacer música y efectos de sonido.

▶ Funciona de la siguiente forma: el VIC reconoce la sentencia For y la variable de control que la sigue (D, que en este caso significa "delay" (retraso)). D asume el valor del primer número después del signo igual a (D=1) y el ordenador busca la siguiente instrucción que, en este caso, es NEXT D. Cuando reconoce esto, el VIC-20 vuelve automáticamente a la sentencia For y la variable D se incrementa por 1. Continúa buscando la siguiente instrucción —NEXT D— hasta que D es igual al número que sigue TO en la sentencia. Este proceso se llama bucle. Cuando D. finalmente queda igual a 100, el ordenador pasa a la siguiente instrucción después de NEXT D y continúa ejecutando el programa.

Este bucle de retraso es un ejemplo de la aplicación más sencilla de la sentencia For...Next en Basic. Se puede llegar a hacer una programación más complicada con un bucle For...Next intercalando instrucciones entre FOR y NEXT y coordinando la variable de control (D) y el rango (1 a 100) con otos aspectos de un programa. Los bucles For...Next también pueden ir anidados, uno dentro de otro. Si a todo esto se suma el hecho de que los bucles For...Next se usan con frecuencia —hay otro en la línea 45 de este programa— y podrás ver que es una herramienta muy útil para la programación en Basic.



Sin embargo, antes de que se encuentre el segundo bucle For...Next en la línea 45, se le indica al ordenador que apague el sonido (POKE S,0). Después de hacer esto, cuenta hasta 50, ahora usando la variable I, para que se produzca un breve silencio entre las notas. Un segundo consiste en 60 "jiffies" (el VIC-20 dispone de un reloj interno) y podrías modificar el número de "jiffies" en los bucles de retraso de este programa, para hacerte una idea de este modo de cronometrar.

Acuérdate, T representa Tono y fue inicializado a un valor de 128 en la línea 10. La sentencia T=T+1 NO es una cuestión algebraica. Es una vieja técnica de programación que antiguamente iba precedida por la palabra LET (Dejar). Si tienes esto en cuenta, entenderás que la sentencia significa "DEJA que el valor de T tenga el valor actual más uno". Esto significa que T tiene un valor de 129 después de la línea 50.

Bucles GOTO

De momento sáltate la línea 60. El orden en que las instrucciones se ejecutan en Basic está controlado por la numeración de la líneas del programa. A veces es necesario que las instrucciones se repitan o que se ejecuten en un orden diferente del que quede establecido por los números de línea. La instrucción de la línea 70 es un ejemplo de otro tipo de bucle, un bucle GOTO, que realiza ambas cosas. El VIC-20 reconoce que el GOTO en la línea 70 significa "vete a la línea 30 y sigue las instrucciones que encuentres allí. "(Se le puede decir a un ordenador que vaya a cualquier número de líneas de un programa, tanto hacia adelante como hacia atrás). Así que el VIC hace el bucle para volver a la línea 30 y suena otra nota. Cada vez que esto ocurre, se le suma uno a T, de modo que el tono de la nota incrementa un poco. El bucle GOTO incluye todo lo contenido entre las líneas 30-70.

La creación de un bucle GOTO es complicada y te puede traer problemas si no cuidas mucho su estructuración. Por ejemplo, sin la línea 60, el bucle creado en la línea 70 sería infinito. Sólo se podría parar interrumpiendo el programa manualmente. En este programa, hay dos cosas que impiden que esto ocurra, y una de ellas no es una instrucción.

Cuando el bucle GOTO se ha ejecutado 128 veces, T se habrá incrementado hasta tener un valor de 256, y cuando el ordenador intenta introducir 256 mediante un poke en el registro de sonido (o cualquier otro registro), se parará la operación y enviará un mensaje de error. Ningún registro de su procesamiento de 8 bits (como el VIC-20) puede funcionar con un número mayor que 255. Pero esto no va a ocurrir ahora gracias a la instrucción de la línea 60.

If... Then de nuevo

Esta sentencia Basic tiene el poder de tomar una decisión. El ordenador comprobará la condición presentada por la sentencia If, y solamente si encuentra que es verdad ejecutará la instrucción que sigue la sentencia Then. Por lo tanto, cada vez que se pasa por el bucle GOTO, el ordenador comprueba para ver si T es mayor que 253. Si no lo es, el ordenador ignora el resto de la línea.

(Posiblemente hayas notado un error en los números. Dije que el máximo valor permitido para un Poke era 255 —y lo es— pero el programa comprueba para ver si T es igual a 253. Intenta sustituir el valor de 253 por 255 para ver lo que ocurre). Cuando T finalmente sea mayor que 253 —cuando T=254— se ejecuta la instrucción que sigue a THEN. El bucle GOTO, que podía seguir ejecutándose indefinidamente, se da por finalizado, y termina el programa.

Intenta modificar partes del programa para producir sonidos diferentes. Prueba borrando la línea 45 (utiliza las teclas del cursor para intercalar los REM detrás de los números de línea, y no tendrás que volver a teclear la línea para volver a activarla; simplemente quita la sentencia REM). Al omitir esto, se oirá un barrido del generador de notas en vez de notas en vez de notas individuales.

Incrementa T en la línea 50 por un número mayor para dejar más espacio entre cada nota; por ejemplo, puedes poner T=T+2. Quita la línea 50 para crear un bucle indefinido en el que se repite la misma nota una y otra vez. (Se da el mismo resultado cambiando la línea 70 a GOTO 10. ¿Sabes por qué?). Además, puedes trasladar el valor del altavoz (S) a otro de los registros de sonido (S=36875, 36876 o 36877) y tomar nota de las diferencias.

Los Pokes y Peeks de Commodore

Tienes que usar los Pokes en el C-64 para que cante el SID. Cuando se utiliza ▶

un Poke en un ordenador Commodore, estás dando una orden explícita y comprensible (para el ordenador), que será realizada cuando pulsas la tecla Return, tanto en modo directo como cuando se ejecuta un programa. El ordenador entiende el formato Poke (posición de memoria), (valor). Acuérdate que existen dos tipos de memoria en un ordenador, RAM (Memoria de Acceso Aleatorio), donde se almacenan las instrucciones como los Pokes, y ROM (Memoria de Solo Lectura), el "cerebro" del ordenador, que seguirá estas instrucciones si las recibe en el formato correcto.

El ordenador también entiende abreviaturas. Puedes utilizar una letra (llamada una constante) que representa una posición de memoria, si lo especificas al principio del programa. Utiliza la letra S, ya que es fácil de recordar, para que represente la primera posición de memoria (o dirección) del SID. Para hacer esto, pones S=5472. Cada vez que S aparezca en el programa (y no vaya entre comillas ni forme parte de una palabra reservada), el ordenador sustituirá el número 54272. Resulta más fácil teclear S a lo largo del programa, y le resulta más fácil al ordenador utilizar la S que 54272 mientras está trabajando.

A partir de ese momento, se añade una S a un número desde 1 a 28, para indicar otra dirección del chip SID: S+4 significa posición de memoria 54276 (54272+4). Esta posición de memoria en particular contiene algunos parámetros importantes para la voz 1. Si haces un Poke en 54276 (representado por S+4) con un valor específico determinado anteriormente (ahora hablaré de esto), puedes activar una o más de cuatro formas de onda, el bit de control, modulación de timbre, sincronización o voz 1. Las opciones que eliges dependen del valor que envías con la instrucción Poke.

Cuando haces un Peek en una posición de memoria, estás haciendo justo lo contrario de un Poke. Le estás pidiendo al ordenador que te diga el valor almacenado en un registro determinado. No se puede hacer un Peek en los registros del SID de 54272-54296 (de S a S+24): se llaman registros de solo escritura. Se puede hacer un Peek en las posiciones de memoria 54297-54300, y se hace de la siguiente forma: PRINT PEEK(54297). Sin embargo, estos registros son solamente de lectura, lo que significa que en ellos no se puede hacer un Poke. Da la casualidad de que el SID es poco normal en este respecto. En la mayoría de las posiciones de memoria se puede hacer un Peek y un Poke.

En este ejemplo (POKE S+4,33), le estás diciendo al C-64 que haga un Poke de un valor de 33 en la posición de memoria 54276. Vamos a ver lo que significa ese 33.



Bits, Bytes y Nibbles

Un byte es lo que hemos estado llamando una posición de memoria, o un registro (54272 es el primer byte del chip de SID). Pero la unidad más pequeña dentro de todos los ordenadores no es un byte, sino un bit, y hay ocho bits en cada byte. (Cuatro bits —medio byte— se llama un nibble).

Mira la tabla del chip SID (Tabla 1). La posición más pequeña de la derecha representa un bit. Si lees de forma horizontal hacia la izquierda, (ocho bits) es un byte. El ordenador organiza los bytes según la dirección de memoria; por ejemplo, 54276 ó S+4. A ver si puedes encontrar este byte en la tabla. Empieza por la izquierda en las Direcciones de Memoria del SID y lee hacia abajo en la columna para la voz 1 hasta que llegues a S+4. Ahora mira la columna del Nombre de Registro; este byte se llama el registro de control. Si sigues a la derecha, puedes ver los ocho bits que componen el byte en 54276.

Observa las voces 2 y 3 en las columnas al lado de S+4. El chip SID se divide en dos tipos de registros: los que afectan una sola voz, y los que afectan todas las voces juntas. La parte superior de esta tabla representa aquellos registros que afectan la voz 1, 2 ó 3. Siete registros afectan cada voz de forma independiente, de modo que se añade siete a cada dirección para la voz 1 (S+4+7=S+11), para llegar al registro de control para la voz 2. Añade otro siete para la voz 3.

La fila superior de posiciones contiene los números utilizados para activar unos bits determinados en el byte que se está direccionando. Estos son los valores determinados antes, ya mencionados, que se introducen mediante un Poke en las posiciones de memoria. Con esta información, puedes interpretar el valor de 33 en la instrucción POKE S+4,33.

En este caso, 33 es la suma de la columna 32 y la columna 1. Se está diciendo al C-64 que quieres que utilice la forma de onda de diente de sierra (un valor de 32) y que active el sonido (con un valor de 1) para la voz 1 (dirección de memoria 54276). POKE S+4,33 le dice todo esto al ordenador. Para desactivar la voz 1, se utiliza un valor de 0 enviando la instrucción POKE S+4,32. Aunque se haya apagado el sonido, se sigue produciendo la forma de onda de diente de sierra; lo que pasa es que tú no lo oyes.

Sistemas Numéricos

Según el sistema binario, el número 33 sería representado como 00100001, donde el 1 significa activado, y el 0 desactivado. No voy a explicar ahora todos los detalles del sistema binario. Es suficiente decir que el sistema binario, el que utilizan los ordenadores, se basa en las potencias de 2. La Tabla 3 demuestra los números de bits de un byte (se aumentan de derecha a izquierda), junto con las correspondientes potencias de 2 y sus equivalentes en el sistema decimal.

Ten en cuenta que los números de los bits son las potencias correspondientes de 2. Calcula 2 elevado a la potencia de 5 ($2^5=2\times2\times2\times2\times2$) y te dará 32, una parte del número decimal utilizado para hacer un Poke en la posición de memoria de nuestro ejemplo. Se activó el bit número 5 introduciendo el número 32 mediante un Poke en el registro. (La exponenciación en el C-64 se realiza con la tecla de flecha arriba. Para conseguir el resultado de 2 elevado a la potencia de 5 en modo directo, teclea PRINT 2 (flecha arriba) 5, y pulsa la tecla Return.)

El número más alto que se puede introducir en un byte mediante un Poke es 255 (128+64+32+16+8+4+2+1 es igual a 255); esto activa todos los bits de este byte. Los números mayores que 255 tienen que ser divididos en dos bytes. Para saber los bits que son activados por el valor de un Poke, resta la máxima potencia de 2 (en forma decimal) que es igual a o menor que el mismo número del Poke. Digamos que el valor del Poke es 133; 133 menos 128 (que activa el bit número 7) deja 5; 5 menos 4 (que activa el bit número 2) deja 1, que activa el bit número 0. Por lo tanto el Poke 133 activa los bits 7, 2 y 0. 133 como número binario es 10000101.

A partir de ahora, cuando veas la instrucción, "para producir una onda de diente de sierra se tienen que activar los bits 0 y 5 en el Registro de Control del SID para la Voz 1, "consulta las Tablas 1 y 3 y conviértelo en POKE S+4,33. Si practicas un poco a lo mejor terminas sabiendo lo que estás haciendo.

MUSICA-C-64

```
10 REM --- C64 MUSICA ---
20 S=54272:FORI=0TO28:POKES+I,0:NEXT
 :REM LIMPIA EL SID
30 POKES+5,3:POKES+6,0:REM ADSR PARA
 VOZ 1
40 POKES+24,15:REM SUBE EL VOLUMEN
50 READ HF,LF:REM LEE HF Y LF DE LOS
 DATAS
60 IF HF=-1 THENPOKES+24,0:END:REM C
OMPRUEBA EL FINAL DE DATAS
70 POKES,LF:POKES+1,HF:REM COLOCA LF
 Y HF
80 POKES+4,33:REM FORMA DE LA ONDA (
33=DIENTE DE SIERRA)
90 FORI=1TO200:NEXT:REM RETARDO
100 POKES+4,32:FORI=1TO10:NEXT:REM P
UESTA A CERO,RETARDO DE SILENCIO
110 GOTO50:REM REPETIRLO DE NUEVO
200 DATA16,195,17,195,18,209,19,239,
21,31,22,96,23,181,25,50
210 DATA26,156,28,49,29,223,31,165,3
3,135
300 DATA-1,-1
```

TABLA 1. Interface de Sonido del C-64

S=54272

Direcciones de Memoria del SID				Chip del SID							
Voz 1	Voz 2	Voz 3	Nombre de Registro	128	64	32	16	8	4	2	1
S	S+7	S+14	Baja Frecuencia	F_7	F_6	F_5	F_4	F_3	F_2	F_1	F_0
S+1	S+8	S+15	Alta Frecuencia	F_{15}	F_{14}	F_{13}	F_{12}	F_{11}	F_{10}	F_9	F_8
S+2	S+9	S+16	Ancho de Pulso - Bajo	PW_7	PW_6	PW_5	PW_4	PW_3	PW_2	PW_1	PW_0
S+3	S+10	S+17	Ancho de Pulso - Alto	—	—	—	—	PW_{11}	PW_{10}	PW_9	PW_8
S+4	S+11	S+18	Control de Registro	Ruido	Pulso	Diente de Sierra	Triangulo	Test	Mod. de Timbre	Sinc.	Puerta
S+5	S+12	S+19	Ataque/Decaimiento (A/D)	A_3	A_2	A_1	A_0	D_3	D_2	D_1	D_0
S+6	S+13	S+20	Sostenimiento/Relajación (S/R)	S_3	S_2	S_1	S_0	R_3	R_2	R_1	R_0
Filtros:		S+21		—	—	—	—	—	FC_2	FC_1	FC_0
		S+22		FC_{10}	FC_9	FC_8	FC_7	FC_6	FC_5	FC_4	FC_3
		S+23	Resonancia	Res_3	Res_2	Res_1	Res_0	Filtro Ext.	Filt 3	Filt 2	Filt 1
		S+24	Volumen y Filtros	3off	HP	BP	LP	Vol_3	Vol_2	Vol_1	Vol_0
Misc:		S+25	Pot X	PX_7	PX_6	PX_5	PX_4	PX_3	PX_2	PX_1	PX_0
		S+26	Pot Y	PY_7	PY_6	PY_5	PY_4	PY_3	PY_2	PY_1	PY_0
		S+27	OSC_3	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
		S+28	$Envoltura_3$	E_7	E_6	E_5	E_4	E_3	E_2	E_1	E_0

TABLA 3. Números de los bits y potencias correspondientes de 2 con sus equivalentes decimales

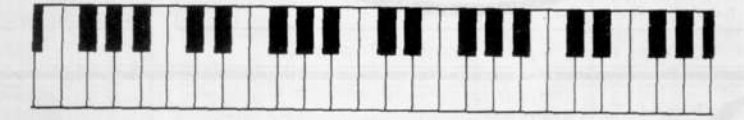
7	6	5	4	3	2	1	0	Número de Bit
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	Potencias de 2
128	64	32	16	8	4	2	1	Valores Decimales

TABLA 2. Posiciones de Memoria de Sonido en el VIC

Posición de Memoria	Nombre de Registro
36874	Voz 1 (rango de C_1 a C_4)
36875	Voz 2 (rango de C_2 a C_5)
36876	Voz 3 (rango de C_3 a C_6)
36877	Ruido blanco
36878	Control de volumen

Términos que deberías saber

- **Las direcciones del SID, o registros, empiezan en la posición de memoria 54272, que será representada por la constante S. Para referirme a los otros registros del SID, añadiré un número a S, de la siguiente forma: S+4, que significa 54276. Un tono generador del C-64 se llama una voz.**
- **La nota, o frecuencia, es lo alto o lo bajo de una melodía en la escala musical, que queda determinada por el número de vibraciones de una onda de sonido durante cada ciclo. El chip SID controla la frecuencia de un sonido utilizando dos registros por voz. Estos registros se llaman de baja frecuencia y de alta frecuencia (S y S+1 para la voz).**
- **La dinámica es la amplitud de sonido: es decir, el volumen. El control del volumen del C-64 (que afecta a todas las voces de la misma forma) se encuentra en S+24, que también controla los filtros. Tiene que contener un número mayor que 0, o de lo contrario no se produce ningún sonido.**
- **El contenido en armónicos de una forma de onda puede ser cambiado usando un filtro, que normalmente reduce el volumen de la voz. Esto se usa generalmente para atenuar una voz resaltando otra. Hay que indicarle al ordenador la voz que se tiene que filtrar y hasta qué grado, utilizando el registro S+23, y fijar los puntos de referencia en S+21 y S+22 (para la voz 1 solamente).**
- **El timbre, o color del tono, se refiere a las cualidades individuales por las que se distingue una voz de otra. Por ejemplo, CONCERT A es una frecuencia específica (440 Hz), que constituye un estándar para afinar los instrumentos de una orquesta. Cada músico afina su instrumento de acuerdo con CONCERT A, nota dada por el primer violín. El tiembre es la diferencia que se parecía cuando un arpa y una trompeta tocan la misma nota. Es lo que hace que nuestras voces sean diferentes.**
- **El control del timbre en el C-64 se realiza mediante los generadores de la envoltura ADSR (dos para cada voz, S+5 y S+6 para la voz 1) que trabajan junto con la forma de onda elegida. ADSR significa Ataque, Decaimiento, Sostenimiento y Relajación. Estas son las características físicas de la envoltura de un sonido.**
- **La instrumentación se describe como textura. Esto puede ser desde un solo de guitarra hasta una orquesta sinfónica. Lo máximo que puedes tener en el C-64 es tres voces sincronizadas, pero puedes engañar tus sentidos para que piensen que se oye más que eso. Se puede empezar, por ejemplo, con un trío de batería, guitarra, y bajo, pero convertir la guitarra en una trompeta. De esta forma piensas que estás oyendo cuatro voces. Esto se consigue con un programa que coordina voces múltiples.**
- **El compás incluye el tempo general de la pieza —rápido, despacio o entre medias— y el ritmo.**



Descubre las nuevas impresoras star para tu Commodore



Star ha lanzado al mercado los nuevos modelos de impresoras para Commodore; doble interface, paralelo centronics y Commodore.
Con este interface para Commodore logramos poder trabajar con todas las grandes prestaciones de esta gama de impresoras, es decir, una gran velocidad, fricción tracción, cinta de máquina de escribir, 100 tipos de letras diferentes...
Además de todas estas prestaciones lograrás con tu Commodore transcribir todos estos tipos de letra con sus propios caracteres gráficos.
Si quieres hacer gráficos trabajando con el Simon's Basic, lograrás hacer Hard copys directamente de pantalla.

Todo esto y mucho más lograrás con las impresoras STAR.

De venta en establecimientos especializados:

IMPORTADO POR:



COMPONENTES ELECTRONICOS, S.A.

08009 BARCELONA. Consejo de Ciento, 409
Tel. (93) 231 59 13

28020 MADRID. Comandante Zorita, 13
Tels. (91) 233 00 94 - 233 09 24

Serenata SID

Muchos de los juegos existentes para el C-64 son divertidos gracias a la excelente música de fondo que tienen, o a los efectos de sonido que, con frecuencia les acompañan.

Cuando los juegos están escritos en Basic no suelen tener música, ya que es bastante difícil de generar y reduce enormemente la velocidad del desarrollo del juego. Aunque la mayoría de las tareas necesarias para la generación del sonido las realiza el chip SID, el programa no sólo tiene que enviarle los cambios de notas, sino que debe controlar la duración de cada una de ellas.

En este artículo se describe un programa en lenguaje máquina que controla los cambios de notas. Su funcionamiento está basado en la estructura de las interrupciones del 64. La CPU es "interrumpida" (deja de ejecutar la tarea que está realizando y ejecuta algunas tareas de gestión interna antes de proseguir lo que estuviera haciendo) sesenta veces por segundo. Estas tareas incluyen la actualización del reloj TI$ y la comprobación del teclado.



M. J. CLIFFORD (RUN EE.UU.)

Adaptado por Alvaro IBAÑEZ

Marcando el Tiempo. El programa "COD-CANC" se añade al conjunto de tareas de gestión interna, por lo que se ejecuta 60 veces cada segundo. Controla el espaciado de las notas mediante dos factores: tempo y duración.

Si el tempo es 4 y la duración 3, la nota sonará durante 12 interrupciones, es decir, 1/5 de segundo. En ese momento, el circuito de esa voz se desactiva, se carga la siguiente frecuencia en el SID y los circuitos vuelven a funcionar. Como esto se realiza con la velocidad del lenguaje máquina, no se aprecia ningún tiempo muerto y el programa Basic se ejecuta a la misma velocidad que si no tuviera música.

El programa que se muestra en el Listado 1 coloca el lenguaje máquina en memoria y lo salva como un fichero de programa en disco o cinta. El programa reside en un bloque de memoria situado encima del Basic, por lo que no se reduce la cantidad de memoria disponible para el programa Basic.

El Listado 2 es un programa en Basic que se utiliza para preparar los datos para las melodías. El programa puede utilizarse tanto para introducir nuevas melodías como para modificar las ya existentes. Cuando se están modificando, se imprimen los valores actuales en la línea de entrada. De esta forma, si quieres dejarlos como están, basta pulsar la tecla Return para reintroducirlos.

Cuando se introduce una melodía nueva, primero el programa pide la envolvente ADSR (Ataque, decaimiento, sostenimiento, relajación) de la primera voz. Se pueden introducir valores entre 0 y 15 para cada uno de los parámetros. Además se puede elegir la forma de onda: 1 para ondas tringulares, 2 para diente de sierra, 3 para pulsos o 4 para ruido. A continuación, el programa te pregunta si prefieres modulación en anillo o sincronización y, si se están utilizando pulsos, la anchura del pulso.

Una vez introducidos los parámetros iniciales de la voz, se piden los valores correspondientes a las notas. Antes de introducir las notas, se puede introducir una clave. Una vez introducida la clave, no es preciso introducir los sostenidos y bemoles habituales de dicha clave.

Las notas se introducen mediante las letras A-G (LA a SOL), el número de octava (de 1 a 7) y su duración. Los sostenidos se indican mediante el símbolo #, mientras que los bemoles han de introducirse mediante sus equivalentes naturales en términos de sostenidos. En la pantalla aparecen los valores de duración para varias notas.

Además de la notación A-G, se puede introducir una R con un valor 0 de octava para el silencio. También se puede introducir CV, 0,1 para cambiar la envolvente ADRS y la forma de onda de la voz durante la ejecución de la melodía. Al final de la melodía se introduce RE, seguido por el número de veces que se desee repetir la melodía (de 0 a 15) y el intervalo que se desea entre las repeticiones.

Cuando se introduce RE, el programa muestra la duración total de la melodía en términos de semicorcheas y de pausas que se hayan introducido. Esto sirve para asegurarse de que las tres voces van sincronizadas. Cuando se termina con la primera voz, se introduce la segunda y luego la tercera.

Las opciones son tuyas

Una vez introducidos los datos, el programa entra en modo modificación, ofreciendo varias opciones. La melodía puede ejecutarse con tempos distintos, cambiando las envolventes iniciales, las notas modificadas o los filtros activados.

Una vez realizadas las modificaciones oportunas, la melodía se salva, junto con el programa en lenguaje máquina que la interpreta, en un fichero de programa. La melodía se puede cargar posteriormente mediante LOAD "melodía", 8,1 e interpretarla con un comando SYS.

C-64

Una vez que comienza la melodía, su ejecución continuará hasta su conclusión mientras que el Basic funciona normalmente. Hay otro comando SYS que sirve para parar la interpretación antes de su terminación, desconectando al "COD.CANC." de las rutinas de interrupción y apagando el SID.

Mediante SYS 49152, la melodía se interpreta con un tempo (por defecto) de 4. Si quieres utilizar un tempo diferente, utiliza el comando POKE780, tempo: SYS 49154. El comando que sirve para detener la ejecución es SYS 49344.

Para ayudarte a comenzar, el Listado 3 contiene los datos de una melodía. Ejecuta este programa después de haber ejecutado el cargador, luego carga y ejecuta el Editor de Melodías del Listado 2. Elige el modo Modificación y usa las distintas opciones para ver cómo se introdujo la melodía.

Por ejemplo, si eliges la opción 1 (Cambiar la Onda), aparecerán simultáneamente en pantalla los valores de las envolventes ADSR y de las formas de onda. Para volver a introducir el valor sin cambios y pasar al siguiente,



pulsa la tecla Return según vaya apareciendo cada valor.

Para interpretar la melodía, elige la opción 4. El tempo apropiado es 4, pero puedes probar con otros valores para comprobar su efecto.

Mediante la opción 5, puedes salvar la melodía junto con el programa en lenguaje máquina, con lo que podrás cargarla posteriormente cuando utilices el Editor. Una vez salvado el programa y la melodía, teclea RUN, elige el modo Modificación y experimenta con distintas formas de onda y filtros. Para realizar cambios, teclea los nuevos valores sobre los anteriores según vayan apareciendo y, a continuación, pulsa la tecla Return.

Añade música a tus programas en Basic sin perder el ritmo

Descripción del cargador Basic

Líneas 10-60: Leen los datos de las líneas 1001-1048 y los coloca en memoria. El último número de cada línea es de comprobación. Si se produce algún error en la introducción de los datos, el programa se parará e informará del número de la línea de datos en la que se ha producido el error.

Líneas 80-110: Informan que la carga del programa se ha realizado satisfactoriamente y ofrece la posibilidad de salvarlo como un fichero de programa.

Línea 200: Asigna el nombre del fichero de programa.

Líneas 210-300: Utiliza la rutina Kernal para salvar el programa en disco. Si se va a utilizar cassette, el 8 que aparece en la línea 260 deberá cambiarse a 1.

Nota: Se pueden omitir las líneas 95-300 si el programa se ejecuta inmediatamente antes de ejecutar el "Editor de Melodías" (listado 2), ya que éste salva el programa en lenguaje máquina junto con los datos. Si decides utilizar en el futuro este procedimiento, asegúrate de que cargas el fichero de la melodía antes de cargar y ejecutar el "SID EDITOR".

Descripción del SID EDITOR

Línea 10: Carga el programa en lenguaje máquina si es necesario.

Líneas 20-55: Inicializa las variables y permite optar entre "nueva canción" o "editar".

Líneas 60-80: Limpia la memoria para albergar una nueva canción y pone el volumen a 15 por defecto.

Líneas 90-100: Introducción de una nueva melodía.

Líneas 330-495: Entrada de la nota, octava y duración.

Líneas 345-415: Decodifica el valor actual obteniendo la nota y octava correspondiente, y la muestra junto con la duración en la línea de entrada de forma que pueda reintroducirlas pulsando la tecla Return.

Línea 420: Entrada de valores.

Líneas 425-435: Comprobación del final, silencio o cambio de ADSR.

Líneas 440-450: Codifica una nota y una octava en un solo byte.

Línea 470: Imprime el total de semicorcheas para mantener las tres voces sincronizadas.

Líneas 500-790: Toma los valores actuales y los escribe en la línea de entrada por si quieres reintroducirlos.

Líneas 350-560: Entrada del ataque y caída (AD), los combina y los coloca en memoria.

Líneas 570-630: Sostenimiento y relajación (SR).

Líneas 640-660: Decodifica la forma de onda actual. R$="S" si se quiere modulación en anillo. S$="S" si se desea sincronización.

Líneas 670-730: Entrada de la onda, modulación en anillo y sincronización.

Líneas 740-780: Si se utiliza una onda en pulso, toma el ancho del pulso.

Líneas 800-890: Menú de opciones para modificación.

Líneas 900-945: Interpreta la melodía.

Líneas 920-940: No se hace nada hasta que el vector de interrupción vuelve a su estado normal cuando termina la melodía, salvo que se pulse la tecla Q, lo que interrumpe la interpretación de la melodía.

Líneas 950-1130: Establecimiento del volumen y de los filtros. Si no se utiliza esta opción, el valor por defecto para el volumen será 15.

Líneas 1600-1695: Inicializa la tabla de valores de las notas según la clave utilizada. Los datos son para clave C (no hay sostenidos ni bemoles). Si se utiliza una clave distinta, no hay que introducir los bemoles y sostenidos normales para dicha clave.

Líneas 1700-1790: Imprime la nota mostrada y los rangos de octavas y duración para varias notas.

Líneas 1800-1890: En modificación, muestra los sostenidos y bemoles de la clave en curso y permite cambiarlos.

Líneas 1900-1920: Final del diálogo.

Líneas 2000 y sucesivas: Salvan la melodía y el programa en lenguaje máquina conjuntamente como un fichero de programa.

Starp

STX - 80

La silenciosa

Con una velocidad de 60 c.p.s en silencio. Standard ASCII, con caracteres internacionales programables. Bloc gráfico y bit image, es la STX - 80 Con un precio tentador Muy fácil de instalar y de programar.

Gemini 10X

La estrella

Nº 1 en el ranking de las impresoras. Con 120 c.p.s. Realmente económica y fiable, ahora compatible con IBM: Gemini 10 Xi. Para descubrir mas cualidades, pruebala tu mismo y las conoceras.

Gemini 15 X

La económica

La impresora para el profesional con un precio mas que razonable. Con una vida del cabezal de mas de 100 millones de caracteres. La mejor relación precio-prestaciones.

Delta 10

La potencia

Interface serie y paralelo. Con un buffer de 8K. Macroinstrucción. Caracteres programables, Impresión bidireccional y lógica. Otras caracteristicas standard. Esto es algo de Delta 10.



arade



COMPONENTES ELECTRONICOS, S.A.

CONSEJO DE CIENTO, 409 - 08009 BARCELONA
TEL. 231 59 13

COMANDANTE ZORITA, 13 - 28020 MADRID
TELS. 233 00 94 - 233 09 24

Delta 15

La potencia ampliada

La potencia con carro ancho, es la impresora Delta 15. Mas de 233 columnas con 160 c.p.s. La justa fuerza. Un esfuerzo inteligente para un precio standard.

Radix 10

La nueva profesional

200 c.p.s con entrada automática de hoja. Un buffer de 16K. Doble velocidad que le permite una letra de calidad. El profesional usa Radix 10. Todas sus prestaciones son de standard.

Radix 15

La nueva profesional ampliada

Superior a lo normal. Con unas prestaciones que la adelantan al futuro. Doble interface. Prevista para resolver todos tus problemas. Esto es Radix 15.

Powertype

La nueva calidad de impresión para tu escritura.

Pon una nueva dimensión a tu correspondencia. Con mas de 100 modelos de margaritas. Una velocidad de 18 c.p.s. Interface serie y paralelo incluidos. Impresión como la de una máquina de escribir superior. Todo esto a un bajo coste y con una gran calidad.

star

The power behind the printed word.

SERENATA SID
CARGADOR BASIC DEL PROGRAMA COD.CANC.

```
1 REM M.J. CLIFFORD
2 REM 2323 W.BANCROFT
3 REM TOLEDO OH 43607
10 REM CARGADOR DE COD.CANC.
15 PRINT"[CLR][7CRSRD][9CRSRR][RVSON]
UN[SPC]MOMENTO.[SPC]POR[SPC]FAVOR"
20 AD=12*4096
30 FORL=1TO48
40 FORX=0TO15:READB:C=C+B:POKEAD+X,B
:NEXT
50 READK:IFC<>KTHENPRINT"ERROR[SPC]E
N[SPC]LA[SPC]LINEA"1000+L:END
60 C=0:AD=AD+16:NEXTL
80 PRINT"[2CRSRD]LA[SPC]RUTINA[SPC]E
N[SPC]C.M.[SPC]ESTA[SPC]EN[SPC]MEMOR
IA[2CRSRD]"
90 PRINT"LO[SPC]PUEDES[SPC]SALVAR[SPC]
CON[SPC]UNA[SPC]CANCION[8SPC]SI[SPC]
LA[SPC]INTRODUCES[SPC]LUEGO.[CRSRD]"

95 PRINT"TAMBIEN[SPC]LO[SPC]PUEDES[SPC]
SALVAR[SPC]SOLO[SPC]PARA[6SPC]PODER[SPC]
CARGARLO[SPC]RAPIDAMENTE";
97 PRINT"[SPC]CUANDO[7SPC]USES[SPC]E
L[SPC][RVSON]SID[SPC]EDITOR[RVSOFF]"

100 INPUT"[CRSRD]LO[SPC]SALVAS[SPC]A
HORA[SPC](S/N)";Q$:IFQ$="N"THENEND
105 IFQ$<>"S"THEN100
110 PRINT"SALVANDO[SPC]EL[SPC]PROGRA
MA[SPC]DE[SPC]C.M.[CRSRD]"
200 F$="COD.CANC"
210 POKE187,PEEK(71):POKE188,PEEK(72
)
220 FA=PEEK(187)+256*PEEK(188)
230 POKE183,PEEK(FA)
240 POKE187,PEEK(FA+1):POKE188,PEEK(
FA+2)
250 POKE251,0:POKE252,192
260 POKE186,8:POKE185,1
270 POKE780,251:POKE781,0:POKE782,19
5
260 SYS65496
290 PRINT"EL[SPC]PROGRAMA[SPC]SE[SPC]
HA[SPC]SALVADO[SPC]COMO:"F$
300 PRINT"[CRSRD]CUANDO[SPC]UTILICES
[SPC]EL[SPC][RVSON]SID[SPC]EDITOR[RVSOFF]
[SPC],[SPC]ESTE"
305 PRINT"[CRSRD]PROGRAMA[SPC]SERA[SPC]
LEIDO[SPC]AUTOMATICAMENTE"
1000 REM -- DATOS C.M. DEL COD.CANC
--
1001 DATA169,4,141,71,194,141,72,194
,169,0,141,4,212,141,11,212,1876
1002 DATA141,18,212,141,24,193,141,1
43,193,141,6,194,173,252,194,141,230
7
1003 DATA21,212,173,253,194,141,22,2
12,173,254,194,141,23,212,169,0,2394

1004 DATA170,157,58,194,232,224,6,20
8,248,169,7,141,70,194,120,173,2371
1005 DATA20,3,141,192,192,173,21,3,1
41,193,192,169,93,141,20,3,1697
1006 DATA169,192,141,21,3,88,173,255
,194,141,24,212,96,206,71,194,2180
1007 DATA240,3,76,191,192,173,61,194
,208,9,173,64,194,141,4,212,2135
1008 DATA32,212,192,173,62,194,208,9
,173,65,194,141,11,212,32,75,1985
1009 DATA193,173,63,194,208,9,173,66
,194,141,18,212,32,194,193,206,2269
1010 DATA61,194,206,62,194,206,63,19
4,173,72,194,141,71,194,173,70,2268
1011 DATA194,208,28,120,173,192,192,
141,20,3,173,193,192,141,21,3,1994
1012 DATA88,169,0,141,4,212,141,11,2
12,141,18,212,141,24,212,76,1802
1013 DATA49,234,169,0,141,24,212,141
,70,194,141,4,212,141,11,212,1955
1014 DATA141,18,212,96,172,58,194,18
5,128,195,141,61,194,190,0,195,2180
1015 DATA224,207,176,22,189,161,194,
188,74,194,141,0,212,140,1,212,2335
1016 DATA173,67,194,141,4,212,238,58
,194,96,224,239,144,27,138,41,2190
1017 DATA15,205,24,193,240,9,238,24,
193,169,0,141,58,194,96,173,1972
1018 DATA70,194,41,6,141,70,194,96,2
,138,41,15,141,3,212,200,1564
1019 DATA185,0,195,141,5,212,185,128
,195,141,6,212,200,185,0,195,2185
1020 DATA141,2,212,185,128,195,41,25
4,141,64,194,9,1,141,67,194,1969
1021 DATA238,58,194,238,58,194,238,5
8,194,208,137,172,59,194,185,128,255
3
1022 DATA196,141,62,194,190,0,196,22
4,207,176,22,189,161,194,188,74,2414

1023 DATA194,141,7,212,140,8,212,173
,68,194,141,11,212,238,59,194,2204
1024 DATA96,224,239,144,27,138,41,15
,205,143,193,240,9,238,143,193,2288
1025 DATA169,0,141,59,194,96,173,70,
194,41,5,141,70,194,96,2,1645
1026 DATA138,41,15,141,10,212,200,18
5,0,196,141,12,212,185,128,196,2012
1027 DATA141,13,212,200,185,0,196,14
1,9,212,185,128,196,41,254,141,2254
1028 DATA65,194,9,1,141,68,194,238,5
9,194,238,59,194,238,59,194,2145
1029 DATA208,137,172,60,194,185,128,
197,141,63,194,190,0,197,224,207,249
7
1030 DATA176,22,189,161,194,188,74,1
94,141,14,212,140,15,212,173,69,2174

1031 DATA194,141,18,212,238,60,194,9
6,224,239,144,27,138,41,15,205,2186
1032 DATA6,194,240,9,238,6,194,169,0
,141,60,194,96,173,70,194,1984
1033 DATA41,3,141,70,194,96,2,138,41
,15,141,17,212,200,185,0,1496
1034 DATA197,141,19,212,185,128,197,
141,20,212,200,185,0,197,141,16,2191

1035 DATA212,185,128,197,41,254,141,
66,194,9,1,141,69,194,238,60,2130
1036 DATA194,238,60,194,238,60,194,2
08,137,96,74,79,61,11,11,11,1866
1037 DATA32,32,32,33,33,33,0,1,1,255
,1,1,2,2,2,2,462
1038 DATA2,2,2,3,3,3,3,3,4,4,4,4,5,5
,5,6,58
1039 DATA6,6,7,7,8,8,9,9,10,10,11,12
,12,13,14,15,157
1040 DATA16,17,18,19,20,21,22,24,25,
27,28,30,32,34,36,38,407
1041 DATA40,43,45,48,51,54,57,61,64,
68,72,76,81,86,91,96,1033
1042 DATA102,108,115,122,129,137,145
,153,163,172,183,193,205,217,230,244
,2618
1043 DATA0,205,233,6,37,69,104,140,1
79,220,8,54,103,155,210,12,1735
1044 DATA73,139,208,25,103,185,16,10
8,206,53,163,23,147,21,159,50,1679
1045 DATA205,114,32,216,156,107,70,4
7,37,42,63,100,154,227,63,177,1810
1046 DATA56,214,141,94,75,85,126,200
```

```
,52,198,127,97,111,172,126,188,2062
1047 DATA149,169,252,161,105,140,254
,194,223,88,52,120,43,83,247,31,2311

1048 DATA210,25,252,133,189,176,103,
0,0,0,0,0,0,0,0,15,1103
1050 END
```

SERENATA SID
SID-EDITOR

```
1 REM SID EDITOR
2 REM M.J. CLIFFORD
3 REM 2323 W. BANCROFT ST
4 REM TOLEDO, OH 43607
6 PRINT"[CLR][4CRSRD][RVSON]CARGANDO
[SPC]LENGUAJE[SPC]MAQUINA"
10 IFPEEK(49152)<>169THENLOAD"COD.CA
N*",8,1
20 PRINT"[CLR][5CRSRD][14SPC][RVSON]
[SPC]SID[SPC]EDITOR[SPC][RVSOFF][2CRSRD]
"
40 FORX=0TO6:READNV(X):NEXT
50 NA=12*4096+256*3:PA=NA-4:DA=NA+12
8
55 INPUT"[2CRSRD]NUEVA[SPC]MELODIA[SPC]
O[SPC]EDITAR[SPC](N/E)";Q$:IFQ$<>"N"
THEN800
60 PRINT"[CRSRD][RVSON]BORRANDO[SPC]
LA[SPC]MEMORIA"
65 FORX=0TO3:POKEPA+X,0:NEXT
70 FORY=0TO2:A=NA+256*Y
75 FORX=0TO255:POKEA+X,0:NEXT:NEXT
80 POKEPA+3,15
90 GOSUB1600
100 FORV=0TO2:N=1:GOSUB500:GOSUB330:
NEXTV:GOTO800
329 REM ** VALORES DE LAS NOTAS **
330 GOSUB1700
335 TT=0
340 FORN=3TO126:PRINT"[HOM][20CRSRD]
NOTA[SPC]#";N-2
345 NT=PEEK(NA+256*V+N):D=PEEK(DA+25
6*V+N):IFNT=0THENNT=1
350 IFNT>239THENN$="RE":OC=NT-240:GO
TO410
355 IFNT>207THENN$="CV":OC=NT-208:GO
TO410
360 IFNT=86THENN$="R":OC=0:GOTO410
365 OC=INT(NT/12):T=NT-OC*12:OC=OC+1
:IFT=0THENT=12:OC=OC-1
370 IFT=1ANDNV(1)=13THENN$="B":OC=OC
-1:GOTO410
372 IFT=10ANDNV(6)=9THENN$="G#":GOTO
410
375 IFT=2ANDNV(1)=14THENN$="B":OC=OC
-1:GOTO410
380 Y=2
385 IFT=NV(Y)THENN$=CHR$(65+Y):GOTO4
10
390 IFT<NV(Y)THENN$=CHR$(64+Y)+"#":G
OTO410
395 IFY>1ANDY<6THENY=Y+1:GOTO385
400 IFY=6THENY=0:GOTO385
405 IFY=0THENY=1:GOTO385
```

```
410 PRINT"[22CRSRR][15SPC][CRSRU]"
415 PRINT"[22CRSRR]"N$","MID$(STR$(O
C),2)","MID$(STR$(D),2)"[CRSRU]"
420 INPUT"NOTA,OCTAVA,DURACION";N$,O
C,D
425 IFN$="RE"THENNT=240+OC:GOTO455
430 IFN$="CV"THENN=N+1:GOSUB500:GOSU
B1700:GOTO465
435 IFN$="R"THENNT=86:GOTO455
440 IFOC<1OROC>7THENPRINT"[2CRSRU]":
GOTO410
445 T=ASC(N$)-65:NT=NV(T)+12*(OC-1):
IFNT<0THENNT=0
450 IFRIGHT$(N$,1)="#"THENNT=NT+1
455 POKENA+256*V+N,NT:POKEDA+256*V+N
,D:TT=TT+D
460 IFNT>239THENN=127
465 NEXTN:POKENA+256*V+127,240:POKED
A+256*V+127,1
470 PRINT"[CLR][5CRSRD]LA[SPC]DURACI
ON[SPC]TOTAL[SPC]DE[SPC]ESTA[SPC]VOZ
":PRINT"[CRSRD]ES[SPC]DE";TT;"SEMICO
RCHEAS"
475 PRINT"[3CRSRD]PULSA[SPC][RVSON][3SPC]
ESPACIO[3SPC][RVSOFF][SPC]PARA[SPC]S
EGUIR"
480 PRINT"[CRSRD]PULSA[SPC][RVSON][SPC]
R[SPC][RVSOFF][SPC]PARA[SPC]VOLVER[SPC]
A[SPC]INTRODUCIR[4SPC]ESTA[SPC]VOZ."
485 GETA$:IFA$=""THEN485
490 IFA$="R"THEN330
495 RETURN
499 REM ** ENVOLVENTE **
500 PRINT"[CLR][5CRSRD]PARA[SPC]LA[SPC]
VOZ[SPC]#";V+1;":[CRSRD]"
510 Z=PEEK(NA+V*256+N):AT=INT(Z/16):
DE=Z-16*AT
520 PRINT"[14CRSRR]"AT"[CRSRU]"
530 INPUT"ATAQUE[SPC](0-15)";AT:IFAT
<0ORAT>15THEN520
540 PRINT"[CRSRD][13CRSRR]"DE"[2CRSRU]
"
550 INPUT"[CRSRD]CAIDA[SPC](0-15)";D
E:IFDE<0ORDE>15THEN540
560 POKENA+V*256+N,16*AT+DE
570 Z=PEEK(DA+N+V*256):SU=INT(Z/16):
RE=Z-16*SU
590 PRINT"[CRSRD][21CRSRR]"SU"[2CRSRU]
"
600 INPUT"[CRSRD]SOSTENIMIENTO[SPC](
0-15)";SU:IFSU<0ORSU>15THEN590
610 PRINT"[CRSRD][18CRSRR]"RE"[2CRSRU]
"
620 INPUT"[CRSRD]RELAJACION[SPC](0-1
5)";RE:IFRE<0ORRE>15THEN610
630 POKEDA+N+V*256,16*SU+RE:N=N+1
640 PV=PEEK(DA+N+V*256):R$="N":IF(PV
AND4)THENR$="S"
650 S$="N":IF(PVAND2)THENS$="S"
660 PV=INT(PV/16):WV=-(PV=2)-2*(PV=4
)-3*(PV=8)+1
670 PRINT"[CRSRD][11CRSRR]"WV"[2CRSRU]
"
680 INPUT"[CRSRD]ONDA[SPC](1-4)";WV:
IFWV<1ORWV>4THEN670
690 IFWV=1THENPRINT"[CRSRD][SPC]MODU
LAR[SPC]EN[SPC]ANILLO[SPC]CON[SPC]LA
[SPC]VOZ";V-3*(V=0);
700 IFWV=1THENPRINT"(S/N)[2CRSRR]"R$
"[3CRSRL]";:INPUTR$
710 PRINT"[CRSRD][SPC]SINCRONIZAR[SPC]
CON[SPC]LA[SPC]VOZ";V-3*(V=0);
720 PRINT"(S/N)[2CRSRR]"S$"[3CRSRL]"
;:INPUTS$
730 PV=2↑(WV-1)*16-4*(R$="S")-2*(S$=
"S"):POKED+A+N+V*256,PV
740 P1=PEEK(NA+256*V+N-2)AND15:P2=PE
EK(NA+N+V*256):PW=256*P1+P2
750 IFWV<>3THEN790
760 PRINT"[25CRSRR]"PW"[CRSRU]"
770 INPUT"ANCHO[SPC]DE[SPC]PULSO[SPC]
(0-4095)";PW:IFPW<0ORPW>4095THEN396
780 P1=INT(PW/256):P2=PW-256*P1
790 POKE(NA+256*V+N),P2:POKE(NA+256*
V+N-2),P1+208:RETURN
799 :
800 PRINT"[CLR][3CRSRD][2CRSRR]ELIGE
[SPC]UNA[SPC]OPCION:[CRSRD]"
810 PRINT"[3SPC]1>[SPC]CAMBIAR[SPC]O
NDA,ETC":PRINT"[3SPC]2>[SPC]CAMBIAR[SPC]
VALORES[SPC]DE[SPC]LAS[SPC]NOTAS"
815 PRINT"[3SPC]3>[SPC]CAMBIAR/COLOC
AR[SPC]FILTRO"
820 PRINT"[3SPC]4>[SPC]INTERPRETAR[SPC]
LA[SPC]CANCION":PRINT"[3SPC]5>[SPC]S
ALVAR[SPC]LA[SPC]CANCION/FINALIZAR[CRSRD]
"
830 INPUT"NUMERO";C
840 IFC<1ORC>5THEN800
850 ONCGOTO870,880,950,900,1900
870 FORV=0TO2:N=1:GOSUB500:NEXTV:GOT
O800
880 INPUT"[2CRSRD]PARA[SPC]QUE[SPC]V
OZ";V:IFV<1ORV>3THEN880
885 GOSUB1800
890 V=V-1:GOSUB330:GOTO800
900 INPUT"[CRSRD]A[SPC]QUE[SPC]TEMPO
";TP
905 PRINT"[2CRSRD]PULSA[SPC][RVSON][SPC]
Q[SPC][RVSOFF][SPC]PARA[SPC]PARAR[SPC]
LA[SPC]CANCION"
910 POKE780,TP:SYS49154
920 IFPEEK(788)=49THEN800
930 GETA$:IFA$<>"Q"THEN920
940 SYS49344:REM PARADA
945 GOTO800
949 REM ** FILTROS **
950 PRINT"[CLR][4CRSRD]FILTROS[2CRSRD]
"
955 Z=PEEK(PA)+8*PEEK(PA+1)
960 PRINT"[16CRSRR]"Z"[CRSRU]"
965 INPUT"CORTE[SPC](0-2047)";CF:IFC
F<0ORCF>2047THEN960
970 CH=INT(CF/8):POKEPA,CF-8*CH:POKE
PA+1,CH
975 Z=PEEK(PA+2):FL=0
980 PRINT"[19CRSRR]"INT(Z/16)"[CRSRU]
"
985 INPUT"RESONANCIA[SPC](0-15)";RS:
IFRS<0ORRS>15THEN980
990 FORV=0TO2:PRINT"[17CRSRR]"CHR$(7
8+11*((ZAND(2↑V))/(2↑V)))"[CRSRU]"
995 PRINT"FILTRO[SPC]VOZ";V+1;"(S/N)
";:INPUTQ$:IFQ$="S"THENFL=FL+2↑V
1000 NEXT:FL=FL+16*RS:POKEPA+2,FL
1010 Z=PEEK(PA+3):M=0
1020 PRINT"[10CRSRR]"CHR$(78+11*((ZA
ND16)/16))"[CRSRU]"
1030 INPUT"PASABAJO";Q$:IFQ$="S"THEN
M=16
1040 PRINT"[11CRSRR]"CHR$(78+11*((ZA
ND32)/32))"[CRSRU]"
1050 INPUT"PASABANDA";Q$:IFQ$="S"THE
NM=M+32
1060 PRINT"[10CRSRR]"CHR$(78+11*((ZA
ND64)/64))"[CRSRU]"
1070 INPUT"PASAALTO";Q$:IFQ$="S"THEN
M=M+64
1080 PRINT"[21CRSRR]"CHR$(78+11*((ZA
ND128)/128))"[CRSRU]"
1090 INPUT"VOZ[SPC]3[SPC]APAGADA[SPC]
(S/N)";Q$:IFQ$="S"THENM=M+128
1100 PRINT"[15CRSRR]"ZAND15"[CRSRU]"
```

```
1110 INPUT"VOLUMEN[SPC](1-15)";VL:IF
VL<1ORVL>15THEN1100
1120 POKEPA+3,M+VL
1130 GOTO800
1599 REM ** AJUSTA TECLAS **
1600 RESTORE:FORX=0TO6:READNV(X):NEX
T
1610 PRINT"[CLR][3CRSRD]AJUSTAR[SPC]
LA[SPC]TONALIDAD[SPC]PARA[SPC]LA[SPC]
CANCION"
1620 PRINT"ENTRA[SPC]LAS[SPC]NOTAS[SPC]
QUE[SPC]SON[SPC]#":PRINT"[SPC]POR[SPC]
EJEMPLO:[SPC]ABD[SPC](0[SPC]SI[SPC]N
INGUNA)[SPC][2CRSRD]"
1630 INPUTK$
1635 IFK$="0"THEN1660
1640 FORZ=1TOLEN(K$):Y=ASC(MID$(K$,Z
,1))-65:NV(Y)=NV(Y)+1:NEXTZ
1660 PRINT"[2CRSRD]INTRODUCE[SPC]LAS
[SPC]NOTAS[SPC]QUE[SPC]SON[SPC]BEMOL
ES[CRSRD]"
1670 INPUTK$:IFK$="0"THENRETURN
1680 FORZ=1TOLEN(K$):Y=ASC(MID$(K$,Z
,1))-65:NV(Y)=NV(Y)-1:NEXTZ
1690 RETURN
1695 DATA11,13,2,4,6,7,9
1700 PRINT"[CLR]VALORES[SPC]DE[SPC]L
AS[SPC]NOTAS:[CRSRD]"
1710 PRINT"[16SPC][SHIFTO][10SPC][COMMG]
"
1715 PRINT"[2SPC]G[COMMT][5SHIFTC]F[6SPC]
[SHIFTO][2SPC]=[SPC]1[4SPC][SHIFTQ][COMMG]
.[SPC]=[SPC]6"
1720 PRINT"5[SPC]E[COMMT][5SHIFTC]D[5SPC]
[SHIFTQ][COMMG]"
1730 PRINT"[COMMZ][SHIFTC]C[COMMT][5SHIFTC]
B[SHIFTC][COMMS]"
1735 PRINT"[2SPC]A[COMMT][5SHIFTC]G[SPC]
[SHIFTB][4SPC][SHIFTO][10SPC][COMMG]
"
1740 PRINT"[2SPC]F[COMMT][5SHIFTC]E[SPC]
4[3SPC][SHIFTQ][COMMG][2SPC]=[SPC]2[4SPC]
[SHIFTW][COMMG][2SPC]=[SPC]8"
1745 PRINT"[2SPC]D[COMMT][5SHIFTC]C[SHIFTC]
[COMMX]"
1750 PRINT"[COMMA][SHIFTC]B[COMMT][5SHIFTC]
A[6SPC][SHIFTO][10SPC][COMMG]"
1755 PRINT"[SHIFTB][SPC]G[COMMT][5SHIFTC]
F[5SPC][SHIFTQ][COMMG].[SPC]=[SPC]3[4SPC]
[SHIFTW][COMMG].[SPC]=[SPC]12"
1760 PRINT"3[SPC]E[COMMT][5SHIFTC]D"
1765 PRINT"[COMMZ][SHIFTC]C[COMMT][5SHIFTC]
B[SHIFTC][COMMS][4SPC][COMMG]"
1770 PRINT"[2SPC]A[COMMT][5SHIFTC]G[SPC]
2[3SPC][SHIFTQ][COMMG][2SPC]=[SPC]4[4SPC]
[SHIFTW][3SPC]=[SPC]16"
1775 PRINT"[CRSRD][3SPC]SILENCIO:[SPC]
R,0,DURACION"
1780 PRINT"CAMBIAR[SPC]ENVOLVENTE:CV
,0,1"
1790 PRINT"[4SPC]FIN:[SPC]RE,NUMERO[SPC]
DE[SPC]REPETICIONES,[13SPC]TIEMPO[SPC]
ANTES[SPC]DE[SPC]LA";
1791 PRINT"[SPC]REPETICION"
1795 RETURN
1800 REM COMPRUEBA TONALIDAD
1810 RESTORE:SK$="":FK$=""
```

```
1820 FORY=0TO6:READX:IFNV(Y)>XTHENSK
$=SK$+CHR$(65+Y)
1830 IFNV(Y)<XTHENFK$=FK$+CHR$(65+Y)

1840 NEXTY:IFSK$=""THENSK$="NINGUNA"

1850 IFFK$=""THENFK$="NINGUNA"
1860 PRINT"TONALIDAD[SPC]ACTUAL:[CRSRD]
":PRINT"SOSTENIDOS:[SPC]"SK$
1870 PRINT"BEMOLES:[SPC]",FK$
1880 PRINT"OK[SPC](S/N)[3SPC]S[3CRSRL]
";:INPUTQ$:IFQ$="S"THENRETURN
1890 GOTO1600
1900 PRINT"[CLR][5CRSRD]QUIERES[SPC]
SALVAR[SPC]LA[SPC]CANCION":INPUT"(S/
N)";Q$
1910 IFQ$="N"THENEND
1920 IFQ$<>"S"THEN1900
2000 REM *** SALVAR CANCIONES ***
2005 F$=""
2010 INPUT"[CRSRD]NOMBRE";F$:IFF$=""
THEN2010
2020 F$=F$+".SNG"
2030 POKE187,PEEK(71):POKE188,PEEK(7
2)
2040 FA=PEEK(187)+256*PEEK(188)
2050 POKE183,PEEK(FA)
2060 POKE187,PEEK(FA+1):POKE188,PEEK
(FA+2)
2070 POKE251,0:POKE252,192
2080 POKE186,8:POKE185,1
2090 POKE780,251:POKE781,0:POKE782,1
98
2100 SYS65496
2110 PRINT"HECHO[CRSRD]":END
```

SERENATA SID
SID-DEMO

```
10 REM DATAS DE CANCION
20 PRINT"[CLR][5CRSRD][9CRSRR][RVSON]
[SPC]UN[SPC]MOMENTO,[SPC]POR[SPC]FAV
OR"
30 AD=49916
35 FORL=1TO47
40 FORX=0TO15:READB:C=C+B:POKEAD+X,B
:NEXT
50 READK:IFC<>KTHENPRINT"ERROR[SPC]E
N[SPC]LA[SPC]LINEA";1000+L:END
60 C=0:AD=AD+16:NEXTL
100 PRINT"LA[SPC]CANCION[SPC]ESTA[SPC]
EN[SPC]MEMORIA.[CRSRD]"
1000 REM *** DATOS ***
1001 DATA0,80,192,15,208,66,127,57,5
6,57,54,86,57,56,57,54,1222
1002 DATA86,57,56,57,62,64,62,62,73,
86,73,62,73,62,64,62,1061
1003 DATA62,73,86,73,62,73,71,86,71,
73,71,69,86,57,56,57,1126
1004 DATA54,86,57,56,57,54,86,57,56,
57,62,64,62,62,73,86,1029
1005 DATA86,64,63,64,62,62,73,86,71,
57,58,71,73,62,57,58,1067
1006 DATA71,58,71,71,55,54,52,71,55,
56,54,56,57,57,54,52,944
1007 DATA50,57,57,62,64,62,73,64,71,
57,73,73,62,64,62,73,1024
1008 DATA64,71,57,62,62,241,0,0,,0,0
,0,0,0,0,0,557
1009 DATA0,0,0,240,0,165,16,2,2,4,4,
4,2,2,4,4,449
1010 DATA4,2,2,4,4,4,4,4,4,4,2,2,4,4
,4,4,56
1011 DATA4,4,4,2,2,4,4,4,2,2,4,4,4,2
,2,4,52
1012 DATA4,4,2,2,4,4,4,2,2,4,4,4,4,4
,4,4,56
1013 DATA4,6,2,6,2,4,4,4,4,4,4,4,4,8
,4,4,68
1014 DATA8,4,4,4,4,4,4,24,4,4,8,4,4,
4,4,4,92
1015 DATA4,16,4,4,4,4,12,4,12,4,16,4
,4,4,4,12,112
1016 DATA4,12,4,16,4,12,0,0,0,0,0,0,
0,0,0,0,52
1017 DATA0,0,0,1,208,9,127,54,53,54,
50,86,54,53,54,50,853
1018 DATA86,86,50,56,55,55,86,57,59,
55,55,55,55,55,55,86,1006
1019 DATA55,57,55,86,53,54,54,86,54,
53,54,50,86,54,53,54,958
1020 DATA50,86,86,50,56,55,55,86,57,
57,57,57,57,55,86,55,1005
1021 DATA55,55,55,54,86,55,54,55,55,
59,45,43,86,43,86,43,929
1022 DATA86,50,52,53,54,50,53,54,54,
45,43,42,86,42,86,42,892
1023 DATA86,42,42,42,40,43,33,43,40,
49,33,49,40,43,33,43,701
1024 DATA40,47,43,86,43,42,42,241,0,
0,0,0,0,0,0,0,584
1025 DATA0,0,0,240,0,0,64,2,2,4,4,4,
2,2,4,4,332
1026 DATA4,4,8,8,4,4,4,2,2,4,4,4,4,4
,4,4,68
1027 DATA4,4,4,4,4,4,4,4,2,2,4,4,4,2
,2,4,56
1028 DATA4,4,4,8,8,4,4,4,4,6,2,6,2,8
,4,4,76
1029 DATA8,4,4,8,12,4,4,4,4,4,4,4,4,
4,4,4,80
1030 DATA4,4,4,4,4,4,4,4,4,4,4,4,4,4
,4,4,64
1031 DATA4,4,4,4,4,4,4,4,4,4,4,4,4,4
,4,4,64
1032 DATA4,4,4,32,4,16,4,12,0,0,0,0,
0,0,0,0,80
1033 DATA0,0,0,1,208,9,127,86,38,86,
33,86,38,86,33,86,917
1034 DATA30,29,28,86,33,86,28,86,33,
86,28,86,33,86,28,86,872
1035 DATA33,86,26,86,33,86,38,86,33,
86,38,86,33,86,30,29,895
1036 DATA28,86,33,86,61,60,61,59,59,
45,28,33,40,43,45,38,805
1037 DATA86,40,86,33,86,28,86,33,86,
40,49,47,33,38,86,45,902
1038 DATA86,38,86,45,86,38,49,47,33,
86,33,43,40,43,33,43,829
1039 DATA40,49,33,86,38,47,33,30,26,
241,0,0,0,0,0,0,623
1040 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0
1041 DATA0,0,0,240,0,0,64,4,4,4,4,4,
4,4,4,4,340
1042 DATA8,8,4,4,4,4,4,4,4,4,4,4,4,4
,4,4,72
1043 DATA4,4,4,4,4,4,4,4,4,4,4,4,4,4
,8,8,72
1044 DATA4,4,4,4,6,2,6,2,4,4,8,4,4,4
,4,8,72
1045 DATA8,4,4,4,4,4,4,4,4,8,8,8,8,4
,4,4,84
1046 DATA4,4,4,4,4,8,8,8,8,56,4,4,4,
4,4,4,132
1047 DATA4,4,4,4,6,2,4,4,4,12,0,0,0,
0,0,0,48
1050 END
```

Por Thomas H. SIMMONDS, Jr. (RUN EE.UU.)

Adaptado por Alvaro IBAÑEZ. Traducido por Valerie SHANKS

Volando con el VIC

A ver si eres capaz de despegar y volar tu avión a través del cañón sin chocar con las paredes ni con los aviones que vienen en dirección contraria.

VIC-20 NO AMPLIADO
CINTA O DISCO

"Volando con el VIC" es un juego tipo "arcade", para jugar con joystick en el VIC-20 no ampliado. Tienes que despegar y volar a través del cañón sin chocar con las paredes ni con los otros aviones. Si llegas a los 2.460 puntos, el mensaje "LO HICISTE" aparecerá en pantalla. Hay cuatro niveles de dificultad. Van desde un juego corto (C) y posible (P) hasta un juego largo (L) e imposible (I). No conozco a nadie que haya llegado al nivel más difícil, dado la forma en que está programado el juego.



Descripción del programa

El programa está escrito en Basic, prestando mucha atención a la estructura al bucle principal del juego para que vaya lo más rápido posible. Dada la memoria limitada del VIC no ampliado, se carga el programa en dos partes:

El Listado 1 presenta las instrucciones e introduce en la memoria, mediante unos pokes, los datos para los 60 caracteres personalizados. Finalmente, los 512 bytes de memoria de caracteres personalizados están protegidos, y el Listado 2, el programa principal, está cargado y ejecutado mediante el uso de unos pokes en el buffer del teclado.

El Listado 2 consiste en tres secciones principales: la inicialización, el aeropuerto y las maniobras en el cañón. La parte del juego dedicada al aeropuerto, una subrutina localizada al final del programa, consiste en la inicialización, las sentencias que dibujan el aeropuerto y las maniobras del avión. La siguiente sección contiene la inicialización del cañón, el bucle de vuelo por el cañón, y las rutinas de choque y explosión y la puntuación.

Para cargar los programas, teclea primero el Listado 1. Si vas a salvar el programa en disco, omite la línea 130 y quita las sentencias REM de las líneas 140 y 150.

Ten cuidado al teclear la sentencia Data, ya que contienen los caracteres personalizados de las rutinas del aeropuerto y del cañón. Salva el programa antes de ejecutarlo.

Segundo, teclea el Listado 2, y sálvalo con el nombre "F0" si estás utilizando la unidad de discos. Si estás utilizando cinta, salva el Listado 2 directamente después del Listado 1. De esa forma, cuando se ejecuta el Listado 1, el Listado 2 se carga y se ejecuta automáticamente.

A continuación se presenta una descripción más detallada del Listado 2 y del bucle del juego.

Inicialización

Los parámetros necesarios para la inicialización se encuentran en tres secciones del programa: al principio, para establecer el nivel de dificultad y el color de la pantalla, y para activar los carac-

teres personalizados; en la subrutina del aeropuerto, para establecer las sentencias que dibujan los caracteres personalizados para crear el aeropuerto, y luego para determinar el sonido, las constantes del joystick y la posición inicial del avión; en la rutina del cañón, para inicializar las dimensiones de los "arrays", los "strings" y la posición inicial de la entrada del cañón y el avión.

En la subrutina del aeropueto, el avión aparece en pantalla, se detecta la dirección del joystick, y el avión se mueve de acuerdo con esto. Se incluyen sentencias If... Then para determinar si el avión se encuentra dentro de los límites de la pantalla, si ha chocado con cualquier cosa en el aeropuerto y, finalmente, si se dirige al sur hacia el cañón. Cuando se cumple esta última condición, la parte del programa que incluye el movimiento por el cañón asume el control del avión. Como verás ahora, el bucle del cañón está programado aparte para que sea más rápido.

El cañón

Para empezar, se realizan unas operaciones de "mantenimiento", incluyendo fijar los "strings" (líneas 100-140) que se utilizan para crear las sentencias aleatorias que determinan la dirección en que se gira el cañón. Toma nota que a diferencia de otros programas de gráficos parecidos que utilizan sentencias de este tipo, los cuatro "strings" E$ se crean utilizando caracteres personalizados iguales en los bordes del cañón.

Los "strings" E$ se eligen utilizando el "array" E(I,J), lo que permite que el cañón tenga unos bordes suaves a medida que pasan las secciones hacia la derecha, la izquierda o recto. La línea 150 utiliza dos sentencias aleatorias —Y, para localizar la entrada del cañón, y X, para colocar el avión—. La sentencia poke de la línea 160 coloca el cursor en la penúltima fila de la parte inferior de la pantalla. Aquí se encuentra la primera sentencia del cañón.

Las líneas 170-270 constituyen el núcleo del juego. Estas forman el bucle que controla el avión, dibuja el cañón y detecta los choques. En estas líneas se han tomado medidas para que este programa en Basic vaya más rápido. Por

El programa no es largo y se presta a modificaciones. Los "strings" en las líneas 100-140 que dibujan el cañón se componen en su mayoría de caracteres gráficos elegidos al azar. A medida que juegas, observa lo que pasa con estas sentencias.

ejemplo, un bucle For...Next ha sido empleado en las líneas 170 y 270; esto resulta más rápido que usar GOTO 170 en la línea 270.

Se ha empleado el comando RND(0) en vez del comando más normal RND(1), y se han usado puntos (.) para sustituir los ceros (0) en las líneas 170, 180, 190, 210, 240, 250 y 260; todo esto aumenta la velocidad del Basic.

Las sentencias If de las líneas 170 y 180 deciden aleatoriamente si el cañón gira hacia la izquierda, la derecha o sigue recto. Las líneas 190 y 200 contienen las sentencias If que mantienen el cañón en pantalla. En la línea 210, el avión baja por la pantalla y se comprueba para ver si llegó a pasar por el cañón. Si quieres un juego más largo o más corto, puedes modificar la variable W en las líneas 20 y 30.

En la línea 220, la primera sentencia calcula Q$, el "string" que dibuja en pantalla; a continuación el color actual del avión se convierte en el color del fondo mediante unos pokes para preparar el movimiento. El "string" Q$ se imprime en la parte inferior de la pantalla y todas las otras líneas son empujadas hacia arriba, dando la impresión de que el avión se está desplazando por el cañón. A continuación, se actualiza la posición del avión y éste se convierte en amarillo mediante unos pokes para que vuelva a ser visible. Finalmente, K asume el valor de X. Se detecta el choque con la pared o con otro avión mediante la sentencia If...Then de la línea siguiente.

La línea 240 utiliza los pokes para sacar aviones de colores al azar. La frecuencia con que aparecen estos aviones está controlada por la variable D, fijada por la opción posible/imposible. Puedes cambiar el juego para que resulte más fácil o más difícil si cambias el valor de la variable D en la línea 20.

Las dos líneas siguientes leen el joystick, modifican X, la posición del avión, y fijan P, el carácter personalizado para que el avión vaya hacia la derecha, izquierda o recto. La última línea del bucle es una sencilla Next y vuelve a enviar el programa por el bucle.

Ultimas Rutinas y Modificaciones

Las líneas 280 y 290 producen los efectos visuales y sonoros del choque. La pantalla tiembla haciendo un poke en la dirección localizada en el centro de la pantalla a la vez que los colores de la pantalla se cambian aleatoriamente. Después de la explosión, se actualiza la puntuación y los comentarios sobre el resultado aparecen en pantalla. El jugador tiene que pulsar fuego para volver a jugar.

El programa no es largo y se presta a modificaciones. Los "strings" en las líneas 100-140 que dibujan el cañón se componen en su mayoría de caracteres gráficos elegidos al azar. A medida que juegas, observa lo que pasa con estas sentencias. Verás que cambian los caracteres individuales. Esto es el resultado de sacar los caracteres gráficos de una parte de la memoria que es el programa en Basic en vez de la ROM de caracteres.

Sin embargo, la entrada al cañón y los caracteres más inmediatos en cada lado son muy importantes. Puedes intentar cambiar la anchura del cañón para que el juego resulte más fácil. Si modificas la línea 240, podrías introducir cualquier obstáculo que no fuese un avión.

Si el juego resulta demasiado difícil, puedes fijar la sentencia RND en otro valor que no sea >0.4 Si lo subes a >0.67, salen más secciones rectas en el cañón, y resulta mucho más fácil ganar. ¡Que te diviertas!

LISTADO 1

```
1 GOTO10
2 FORI=1TO63:X=X+10:PRINT"[RVSON]"15
0+X"[RVSOFF]";:FORJ=1TO8:REHDA:PRINT
A;:NEXT:PRINT:NEXT
10 REM VOLANDO CON EL VIC(CARGADOR D
E GRAFICOS)-POR T.SIMMONDS
20 PRINT"[CLR][2CRSRD][2SPC][RVSON]V
OLANDO[SPC]CON[SPC]EL[SPC]VIC[RVSOFF]
"
```

```
30 PRINT"[CRSRD][4SPC]INSTRUCCIONES"
:PRINT"[CRSRD][SPC]EL[1]E[SPC][RVSON]
C[RVSOFF]ORTO[SPC]O[SPC][RVSON]L[RVSOFF]
ARGO"
40 PRINT"[CRSRD][SPC]USA[SPC]EL[SPC]
JOYSTICK"
60 PRINT"[CRSRD][SPC]PARA[SPC]SALIR[SPC]
DEL":PRINT"[CRSRD][SPC]AEROPUERTO[SPC]
Y"
70 PRINT"[CRSRD][SPC]DIRIGETE[SPC]AL
```

```
..." PRINT"[CRSRD][3SPC][RVSON][GRN]
GRAN[SPC]CAN[CRSRL][CRSRU][RVSOFF][SHIFTR]
[RVSON][CRSRD]ON[RVSOFF]"
80 PRINT"[CRSRD][2SPC]PULSA[SPC]UNA[SPC]
TECLA":PRINT"[CRSRD][SPC]PARA[SPC]CA
RGAR[SPC]LOS":PRINT"[CRSRD][SPC]GRAF
ICOS"
90 GETA$:IFA$=""THEN90
100 PRINT"[CLR][8CRSRD][2SPC]CARGAND
O[SPC]GRAFICOS"
110 FORI=0TO511:READCC:POKE7168+I,CC
:NEXT
120 POKE56,28:POKE55,0:CLR
130 POKE198,1:POKE631,131:NEW:END
140 REM POKE198,10:POKE631,76:POKE63
2,111:POKE633,34:POKE634,70:POKE635,
48
150 REM POKE636,34:POKE637,44:POKE63
8,56:POKE639,58:POKE640,131:END
155 REM SI TIENES UNIDAD DE DISCOS,
BORRA LA LINEA 130 Y QUITA LOS 'REM'

156 REM DE LAS LINEAS 140 Y 150.
157 REM LA SEGUNDA PARTE HA DE SER G
RABADA CON EL NOMBRE 'F0'
159 STOP
160 DATA1,84,210,216,254,240,254,124

170 DATA102,60,153,219,255,126,60,24

180 DATA128,42,75,27,127,15,127,62
190 DATA128,192,224,224,224,192,192,
128
200 DATA1,3,7,7,15,7,3,1
210 DATA255,255,254,252,224,192,128,
128
220 DATA128,192,224,224,240,254,255,
255
230 DATA255,255,127,63,7,3,1,1
240 DATA1,1,7,31,31,63,255,255
250 DATA255,254,252,248,252,252,254,
255
260 DATA255,127,127,255,127,63,127,2
55
270 DATA8,127,62,8,8,8,28,8
280 DATA24,60,126,255,219,153,60,102

290 DATA96,96,0,0,0,0,0,0
300 DATA28,57,115,254,54,115,57,28
310 DATA1,84,210,216,254,240,254,124

320 DATA0,0,0,122,66,114,66,67
330 DATA56,156,206,127,127,206,156,5
6
340 DATA128,42,75,27,127,15,127,62
350 DATA0,0,0,34,28,8,8,?00
360 DATA49,74,67,74,50,0,0,0
370 DATA146,90,214,82,82,0,0,0
380 DATA139,116,36,36,35,0,0,0
390 DATA36,180,172,164,36,0,0,0
400 DATA0,0,4,10,8,10,6,0
410 DATA0,0,196,170,174,202,170,0
420 DATA252,254,240,254,216,210,84,1

430 DATA1,1,1,1,1,1,1,1
440 DATA255,0,0,0,0,0,0,0
450 DATA62,127,15,127,27,75,42,128
460 DATA0,0,36,52,44,36,0,0
470 DATA0,14,16,12,2,28,00
480 DATA0,0,0,0,0,0,0,0
490 DATA255,255,255,255,255,255,255,
255
500 DATA224,64,104,111,124,47,41,7
510 DATA12,12,0,0,0,0,0,0
520 DATA31,32,64,255,2,36,104,240
530 DATA248,4,2,255,2,36,104,240
540 DATA60,66,141,141,145,161,66,60
550 DATA0,7,9,9,127,255,255,255
560 DATA0,24,254,255,255,255,255,255

570 DATA0,0,68,84,84,124,0,0
580 DATA0,30,32,16,8,16,32,30
590 DATA0,0,30,32,64,64,32,16
600 DATA16,32,64,64,32,30,0,0
610 DATA0,60,66,64,64,64,60,2
620 DATA2,2,2,66,60,0,0,0
630 DATA114,162,114,2,151,146,96,0
640 DATA130,114,130,2,114,130,242,2
650 DATA112,130,242,2,98,146,98,2
660 DATA28,34,34,28,32,16,30,16
670 DATA64,32,60,32,64,0,64,124
680 DATA0,0,0,0,0,0,0,255
690 DATA128,128,128,128,128,128,128,
128
700 DATA64,0,124,0,40,68,68,56
710 DATA0,0,0,62,8,16,62,0
720 DATA0,0,150,213,181,149,150,0
730 DATA36,0,18,36,0,72,36,0
740 DATA0,0,0,0,24,36,24,24
750 DATA40,40,30,0,20,34,34,28
760 DATA32,0,62,8,16,62,0,30
770 DATA0,0,32,208,208,32,0,0
780 DATA0,0,4,11,11,4,0,0
790 DATA24,24,36,24,0,0,0,0
```

LISTADO 2

```
20 D=30:W=12:CU=36869:SYS65017
30 PRINT"[CLR][3CRSRD][3SPC][RVSON]C
[RVSOFF]ORTO[SPC]O[SPC][RVSON]L[RVSOFF]
ARGO",:INPUTL$:IFL$="L"THENW=22
40 PRINT"[CRSRD][RVSON]P[RVSOFF]USIB
LE[SPC]O[SPC][RVSON]I[RVSOFF]MPOSIBL
E":INPUTP$:IFP$="I"THEND=10
50 S1=36877:POKECU,255:POKES1+2,110:
L=30720:P=1:N=0:S=7680
60 GOSUB390
70 S=7680:T=7702:C=38400:R=37139:IFF
=1THEN150
80 DIME$(4):DIME(4,2):FORI=0TO4:FORJ
=0TO2:READE(I,J):NEXTJ:NEXTI
90 DATA0,4,2,1,2,4,0,4,2,0,0,0,1,2,4

100 E$(0)="[SHIFTH][COMMQ][SHIFTD][COMMW]
[SHIFTS][COMMR][SHIFTE][COMMT][SHIFTY]
[COMMY][SHIFTO][COMMU][SHIFT*][COMMO]
[SHIFTT][COMMP][SHIFTE][BLK]JI[4SPC]
DI[CYN][SHIFTB][COMMZ][SHIFTX][COMMX]
[SHIFTZ][COMMC][SHIFTP][COMMB][SHIFTU]
[COMMN][SHIFTF][COMMM][SHIFTL][COMML]
[SHIFTW][COMMK][SHIFTD][COMMH]"
110 E$(1)="[COMMZ][SHIFTS][SHIFTX][SHIFTF]
[COMMC][SHIFTJ][COMMV][SHIFTL][COMMB]
[SHIFTU][COMMN][SHIFTI][COMMM][SHIFTR]
[COMMA][SHIFTE][COMMD][BLK]JC[4SPC]J
1[GRN][SHIFTV][COMMF][SHIFTM][COMMG]
[SHIFTB][COMMH][SHIFTC][COMMK][SHIFTF]
[COMML][SHIFTA][COMMQ][SHIFTQ][COMMW]
[SHIFTG][COMME][SHIFTD][COMMR]"
120 E$(2)="[COMMG][SHIFT+][COMMH][SHIFT£]
[COMMJ][SHIFT-][COMMK][SHIFTL][COMML]
[SHIFTJ][COMMA][SHIFTG][COMMS][SHIFTF]
[COMMD][SHIFTS][COMMF][BLK]JF[4SPC]G
F[CYN][SHIFTB][COMML][SHIFTC][COMMR]
[SHIFTX][COMME][SHIFTS][COMME][SHIFTG]
[COMMQ][SHIFTH][COMM*][SHIFTT][COMM@]
[SHIFTE][COMMU][SHIFTW][COMMN]"
130 E$(3)=""
140 E$(4)="[COMMP][SHIFTM][COMMO][SHIFTN]
[COMMI][SHIFTV][COMMU][SHIFTX][COMMY]
[SHIFTL][COMMT][SHIFTJ][COMMR][SHIFTG]
[COMME][SHIFTD][SHIFTE][BLK]JE[4SPC]
```

```
HI[GRN][COMMR][COMMG][SHIFTC][COMMR]
[SHIFTN][COMME][SHIFTV][COMMH][SHIFTX]
[COMMD][COMMQ][COMMN][SHIFT+][COMMC]
[SHIFT-][COMMW][SHIFT£][COMMY]"
150 Y=INT(RND(0)*12+6):X=RND(0)*20+4
6:V=0:Q=0:M=6:U=8164
160 POKE214,22:PRINT
170 FORI=1TO500:IFRND(.)>.4THENV=E(V
,2):Y=Y+V-3:GOTO190
180 V=E(V,INT(RND(.)*2))
190 IFY<5THENY=5:V=.
200 IFY>18THENY=18:V=1
210 Q=Q+1:IFQ>WTHENQ=.:X=X+22:IFX>46
7THEN320
220 O$=MID$(E$(V),Y,23):POKEC+K,6:PR
INTQ$:POKES+X,P+1:POKEC+X,7:K=X
230 N=N+1:IFPEEK(T+X)<>32THEN280
240 IFRND(.)*D<1THENPOKEU-Y,11:POKEU
+L-Y,RND(.)*6
250 P=.:POKER,.:POKEH,.:IF((PEEK(Z)A
ND128)=.)THENX=X+1:P=P+1
260 POKEH,255:IF((PEEK(PA)AND16)=.)T
HENX=X-1:P=P-1
270 NEXT
280 FORI=15TO0STEP-.3:POKES1,RND(0)*
100+100:POKES1+19,RND(0)*20+2:POKES1
+1,I
290 POKES1+2,RND(0)*255:NEXT:POKES1,
0:POKES1+2,110:POKES1+19,13
300 IFF1=1THENF1=0:RETURN
310 POKECU,242:PRINT"[CLR][4CRSRD][4SPC]
CHOCASTE[SPC]Y[SPC]...":PRINT"[CRSRD]
[7SPC][RVSON][GRN]EXPLOTASTE[CYN]":G
OTO330
320 POKECU,242:POKES1+1,0:PRINT"[CLR]
[4CRSRD][2SPC]FELICIDADES[SPC]!!!":P
RINT"[5SPC]LO[SPC]HICISTE"
330 SS=10:IFD=10THENSS=15
340 IFN>N1THENPRINT"[CRSRD][SPC]NUEV
O[SPC][RVSON][PUR]RECORD";N*SS:PRINT
"[CRSRD][4SPC][RVSON][YEL]RECORD:[RVSON]
[BLK]"N1*SS:N1=N:GOTO360
350 PRINT"[CRSRD][2SPC]TU[SPC]PUNTUA
CION:[SPC][RVSON][GRN]";N*SS:PRINT"[CRSRD]
[4SPC][RVSON][YEL]RECORD:[RVSON][BLK]
"N1*SS
360 PRINT"[2CRSRD][CYN]PULSA[SPC][RVSON]
FUEGO[RVSOFF][SPC]PARA[SPC]JUGAR[WHT]
"
370 IF((PEEK(PA)AND32)<>0)THEN370
380 F=1:GOTO50
390 PRINT"[CLR][SPC][GRN]!!!!!!![3SPC]
[GRN][[SPC][CYN]7[GRN][5SPC]5":PRINT
"[SPC]![SPC][CYN]RPSO[GRN][SPC]![3SPC]
[[SPC][CYN]2[GRN][3SPC][CYN]3[GRN][SPC]
5"
400 PRINT"[SPC]![SPC][CYN]XY8[2SPC][GRN]
![3SPC][GRN][[SPC][CYN]<[GRN][3SPC][CYN]
6[GRN][SPC]5":PRINT"[SPC]![SPC][CYN]
TUVW[SPC][GRN]![3SPC][GRN][[SPC][CYN]
;[GRN][5SPC]5"
410 PRINT"[SPC]!!!!!!![BLK][SPC]<[RED]
<[GRN][44444445":PRINT"[10SPC][BLK]M
#[GRN][[2SPC]5[2SPC]5[SPC]5"
420 PRINT"[CYN][SPC]:[SPC][WHT]Q[CYN]
[SPC]:[SPC]:[SPC][PUR]R[CYN][SPC]:[SPC]
:[SPC][BLK]!!!":PRINT"[BLK][!!!!!!!
!!!!!!!!!![CYN]="
430 PRINT"[BLK][![YEL]+[BLK]!!!!!!!
!!!!!!!!":PRINT"[BLK][![YEL],[BLK]!!
!!!!!!!!!!!!!![CYN]="
440 PRINT"[BLK][!!!!!!!!!!!!!!!!!!!"
PRINT"[CYN][SPC]?[SPC]?[SPC]?[SPC]?[SPC]
?[SPC]?[SPC]?[BLK]!!!![CYN]="
450 PRINT"[14SPC][BLK]!!!!!":PRINT"[6SPC]
[WHT]↑[3SPC][PUR]$%[SPC][CYN]>[BLK]!
!!!![CYN]="
460 PRINT"[5SPC][WHT]*&)[2SPC][CYN]!
![SPC][BLK][SPC]!!!!!"
470 PRINT"[6SPC][WHT]←[3SPC][CYN]!![SPC]
[CYN]>[BLK]!!!!![CYN]=":PRINT"[SPC][PUR]
1[11SPC][BLK][SPC]!![YEL]-[BLK]!!"
480 PRINT"[SPC][PUR]0[11SPC][CYN]>[BLK]
!![YEL].[BLK]!![CYN]=":PRINT"[SPC][PUR]
/[11SPC][BLK][SPC]!!!!!"
490 PA=37137:Z=37152:H=Z+2:A=S+149:P
OKES1+1,5:POKES1,160:A1=33:C1=0:A2=A
:B=0
500 A=A+B:POKEA2,A1:POKEA2+L,C1:A2=A
:A1=PEEK(A):C1=PEEK(A+L)
510 IFPEEK(A)>35ANDPEEK(A)<41THENN=N
-10:F1=1:GOSUB280:GOTO390
520 POKEA,P:POKEA+L,7
530 P2=P:POKEH,127:G=(NOTPEEK(Z-1))A
ND60-((PEEK(Z)AND128)=.):POKEH,255
540 B=-22*((GAND8)>.)+22*((GAND4)>.)
-((GAND1)>.)+((GAND16)>.)
550 P=-1*((GAND8)>.)-17*((GAND1)>.)-
14*((GAND16)>.)-12*((GAND4)>.):IFB=.
THENP=P2
560 IFA<7680THENA=7680+INT(RND(0)*40
)
570 IFA>8100THENPRINT"[CLR]":RETURN
580 GOTO500
```

Musicalc. Un buen paquete de programas musicales para el C-64

Una de las principales características del C-64 es su capacidad de sintetizar sonidos. De ello se encarga el SID (Sound Interface Device), un chip que controla todas estas funciones.

No obstante, la programación de música en el C-64 no es nada fácil. No existen comandos en BASIC para ello y el método de los POKES resulta complicado y aburrido —además de difícil de entender.

No es extraño pues que las casas de software hayan diseñado programas musicales, desde los más sencillos hasta los más complicados, e incluso accesorios como teclados de órgano para conectarlos directamente al ordenador.

Características generales

El hardware necesario para utilizar este paquete de programa es un Commodore-64 y una unidad de discos. La impresora es opcional, pero conviene tenerla si se usa el MUSICALC2.

El paquete de programas, diseñado por WAVEFORM y comercializado en España por IDEALOGIC está compuesto por tres MUSICALCs: el Sintetizador y Secuenciador, el "Scorewriter" (escritor de partituras), y el "Keyboard maker" (configurador del teclado). También se incluyen el "Musicalc african/latin rhythm template" y el "Musicalc new wave & rock template2" que contienen acompañamientos y melodías preprogramadas.



Musicalc 1

Es el programa principal. Emite la música y controla la forma de las ondas, los filtros, voces, etc. Se trata de un programa híbrido, es decir, medio basic medio código máquina.

La parte más característica del MUSICALC es el PANEL DE CONTROL. Consta de las siguientes partes:

A la derecha se encuentra la "rejilla", en la que aparecen las tres voces (con distintos colores) moviéndose de iz-





quierda a derecha. Esta rejilla está compuesta de 15 filas de 16 columnas cada una. El total de notas programables, teniendo en cuenta que las treinta y dos melodías diferentes de que se dispone pueden encadenarse, es de 7.680 notas.

En la parte superior izquierda están los controles de las tres voces, independientes entre sí. Cada uno de ellos puede variar el tipo de onda y el tipo de envolvente (ataque, decaimiento, sostenimiento, relajación).

En la parte inferior derecha se encuentran los interruptores para conectar/desconectar los filtros, el tipo de filtro que se utiliza (pasabajos, pasabanda, pasaaltos, desconexión, voz uno, voz dos, voz tres y externo) y debajo de ellos los controles temporizadores, para variar la velocidad de la música.

En la parte inferior derecha hay dos indicadores: "sound" y "score". El primero de ellos indica el tipo de sonido, es decir la posición de los controles de la parte izquierda de la pantalla. El segundo es el indicador del número de melodía que se está ejecutando en ese momento.

Todos los controles de la parte izquierda de la pantalla son del tipo "mando deslizante", es decir, que se pueden mover arriba y abajo. Para regularlos hay que usar las teclas de función.

Además de estos controles deslizantes están los "interruptores" —se usan para seleccionar el tipo de onda, por ejemplo— y se conectan y desconectan del mismo modo.

El acceso a cada uno de estos tipos de controles se realiza pulsando primero la tecla "flecha hacia arriba" para entrar en el panel de control y a continuación pulsando una de las teclas de la parte izquierda del ordenador. Las teclas se corresponden con la posición que ocupan los controles en la pantalla.

La elección entre mandos deslizantes o interruptores se realiza con las teclas del cursor.

Para realizar operaciones a las que no se tiene acceso desde el panel de control —cargar una melodía, por ejemplo— hay que pulsar simultáneamente las teclas SHIFT y RETURN, con lo que nos encontraremos ante el menú de opciones.

Este menú ofrece muchas posibilidades, como borrar canciones, listar el directorio del disco, etc.

El manual de instrucciones

El único programa que se suministra con manual es el MUSICALC1. El dos y el tres carecen de él, pues es innecesario: en cualquier momento podemos pedirle al ordenador ayuda (help) y nos responderá con una información concreta y precisa. Los dos discos de acompañamiento, el "New wave & rock template2" y el "African/Latin rhythm template" tampoco lo llevan, pues únicamente contienen datos de melodías preprogramadas.

Tanto el manual como los textos que aparecen en la pantalla del ordenador están en inglés, aunque sabemos que existe una versión en castellano, con los programas también traducidos que es la que comercializa IDEALOGIC.

Las setenta y dos páginas que componen el manual comienzan hablando de lo que es el MUSICALC, de la garantía, de las precauciones que se han de tomar al salvar programas en el disco y de la forma de hacer funcionar el programa.

A continuación viene un esquema del panel de control con indicaciones sobre cada una de sus partes. También habla sobre los conceptos básicos para programar música y de las posibilidades del programa.

Es realmente sencillo aprender cómo programar música, pues el manual te indica paso a paso las teclas que tienes que pulsar y lo que pregunta y responde el ordenador.

También contiene un sumario de características, una especie de glosario palabra por palabra.

Esta es sin duda la parte más completa del manual, explica qué es y para qué sirve el comando/función/tecla/palabra clave, cómo se puede acceder —desde el teclado, el menú, el panel— y, explicaciones sobre lo que es la forma de una onda, los programas externos que lleva (como el E.DOS), los tipos de filtro y las palabras clave.

Lo último que aparece son unas notas sobre los controles más importantes, con un breve resumen sobre los comandos del panel de control, del menú y sobre los programas externos.

Musicalc2. "Scorewriter"

Este programa viene sin manual de instrucciones, pero la verdad es que no es necesario. Sirve para escribir pentagramas, ya sea en la pantalla o en la impresora.



Con la impresora hemos tenido algún problema. El programa está diseñado para funcionar sólo con la COMMODORE 1525 o con una EPSON con interface CARDCO. Como nosotros no tenemos ninguna de las dos, no hemos conseguido sacar ninguna partitura impresa, pero tenemos entendido que la calidad gráfica es excelente.

Para hacer aparecer la partitura por la pantalla, hay que ajustar primero unos valores, tales como el número de notas que van a aparecer en cada línea, el número de "páginas" que se van a imprimir, etc.

Este programa necesita al MUSICALC1 para funcionar, por lo que resulta inútil por sí solo.

Musicalc3, el "Keyboard maker"

Sirve para crear configuraciones musicales en el teclado de acorde con las necesidades de cada uno. Viene acompañado de unos 30 "teclados preprogramados", es decir configuraciones de teclado que van desde la música clásica hasta el rock..



Este programa también necesita el MUSICALC1 para funcionar.

Los dos discos de acompañamiento, tanto el de rock como el de ritmos latinos y africanos dependen completamente del MUSICALC1. Contienen canciones pequeñas y sonidos que pueden ser alterados desde el panel del MUSICALC1 para lograr nuevos efectos, al igual que con los programas de demostración que vienen con el MISICALC1.

Conclusiones

Waveform ha creado un conjunto de programas que aprovechan perfectamente las posibilidades del 64. Los tres son de una calidad excepcional, especialmente el primero. Es de suponer que Waveform continuará con la "saga" de los MUSICALC, tanto con nuevos programas como con discos de acompañamiento.

El manual es ejemplar. Todos los detalles de los programas aparecen claramente explicados y con un buen número de ejemplos.

El único defecto que le hemos encontrado es que no puede usarse con la impresora MPS-801, tan popular entre los usuarios de Commodore.

Por último decir que, si bien este paquete de programas podría ser usado por cualquiera, está mas bien dirigido a la gente que tiene algunos conocimientos de música, a los estudiantes o a los profesionales.

Este paquete musical ha sido diseñado y programado por Richard Wolton, Bill Moulton, John Shepard, Michael Miller y Cris Grigg. La música de demostración y acompañamientos fue creada por Tom Weiser y Harry Likas.

Todos estos programas los comercializa en España IDEALOGIC.

El precio del paquete completo es de 28.550 ptas.

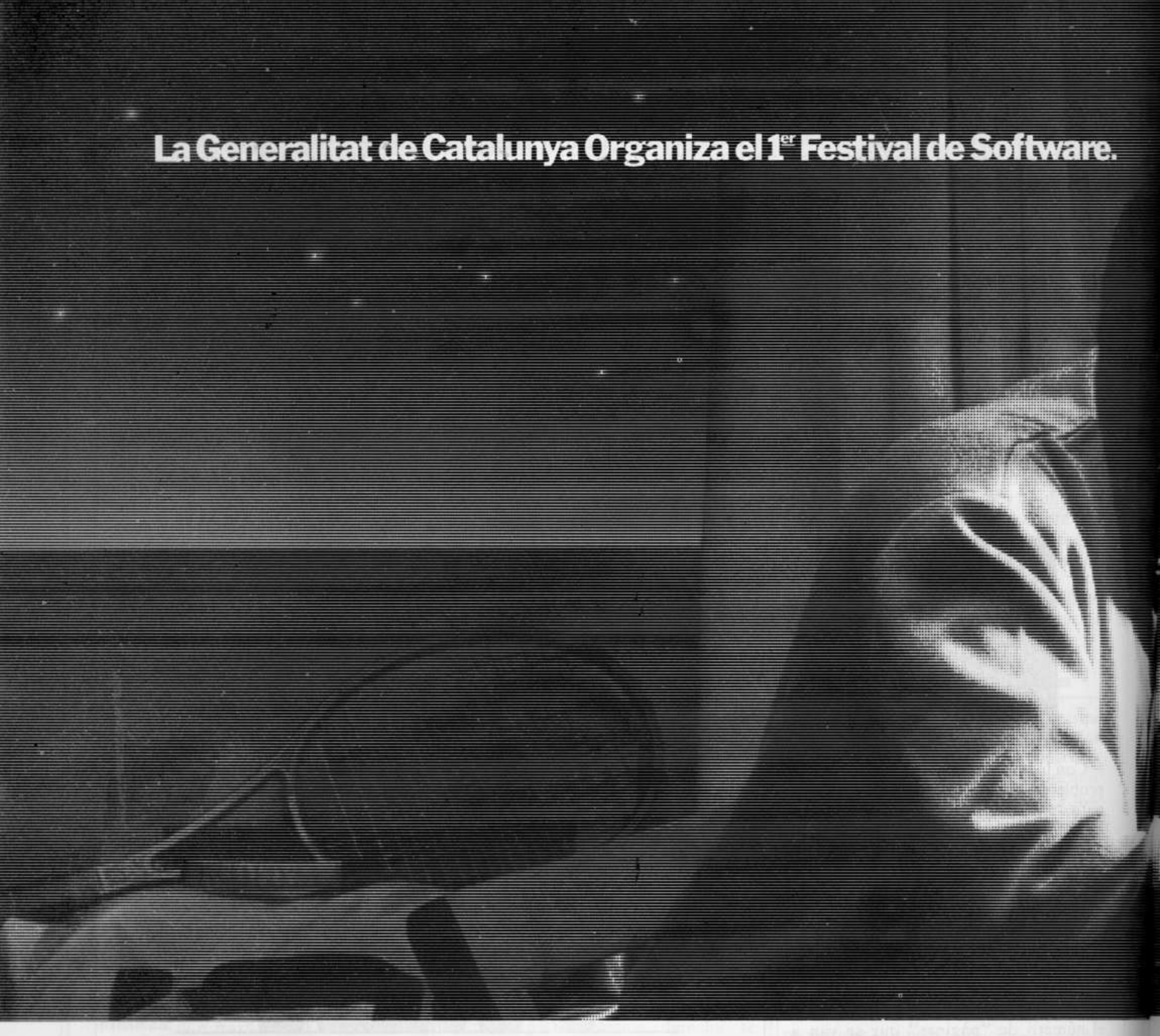
También los venden sueltos al precio de 9.869 el MUSICALC1 y 7.105 el MUSICALC2 y MUSICALC3.

Los TEMPLATES van incluidos en el paquete completo, pero por separado valen 5.487 ptas. cada uno.

Para los que queráis poneros en contacto con IDEALOGIC, su dirección es:
IDEALOGIC, S.A.
c/ Gran Vía de Carlos III, 97 K.
08028 Barcelona
Y su teléfono el (93) 330 33 08

Música para todos

Supongo que si todo el mundo es sincero tendrá que confesar que en algún momento se habrá imaginado subido a un escenario tocando un instrumento de teclado con una soltura y aire de despreocupado que deja asombrado al público. De verdad, siempre ha quedado un poco sospechoso la facilidad con que el teclado de turno emite sonidos, ritmos, acompañamientos, etc., mientras que la estrella de turno le presta poca atención y, eso sí, siempre



Hoy, en solitario.

Dentro de muy pocos años, la informática será un instrumento imprescindible para el conocimiento y el desarrollo de la sociedad. Y para construir este futuro, es necesario que hoy surjan, se promocionen y se ponga de relieve la labor silenciosa y en solitario de muchos nuevos valores que con toda seguridad existen.

Para hacerlo posible, la Generalitat de Catalunya organiza el 1er Festival de Software. A él pueden enviar sus trabajos todas aquellas personas o grupos que hayan creado programas informáticos.

La fecha límite para la presentación de programas, será el 30 de Mayo, y en Octubre, se exhibirá públicamente los seleccionados, que serán puntuados por el público, con cuyos criterios y los del jurado se entregarán los diversos premios, que incluyen una categoría juvenil.

Participa. Y ojalá éste sea tu primer paso hacia el éxito.

MMLB

Mañana, el éxito.

Deseo toda la información

Deseo recibir las bases para participar en el 1er Festival de Software

Tipo de programas que desearía presentar. Programas en ☐ catalán ☐ castellano

Ordenador necesario (marca y modelo) ____________

Pantalla ☐ Color ☐ B/N Impresora (si es necesaria) ____________

Otras periferias o extensiones ____________

Nombre y apellidos ____________ Edad ______

Dirección ____________ Ciudad ____________

Envíe este cupón al Centre Divulgador de la Informàtica. Ap. de Correos 5185. Barcelona.



GENERALITAT DE CATALUNYA
CENTRE DIVULGADOR DE LA INFORMÀTICA

EL CENTRE DIVULGADOR DE LA INFORMÁTICA ORGANIZA ESTE I FESTIVAL DE SOFTWARE EN COLABORACIÓN CON LA ASOCIACIÓN DE TÉCNICOS DE INFORMÁTICA E INFORMAT FERIA DE BARCELONA.

con un solo dedo. Se imagina que una máquina así de completa supone un desembolso muy fuerte por lo que sorprende un poco saber que todo esto lo tienes disponible en tu C-64.

A continución vamos a comentar unos programas que nos han dejado realmente impresionados y que recalcan una vez más el respeto que le tenemos a nuestro querido C-64. Estos programas son instructivos y divertidos a la vez, para el estudiante de música o para el principiante que siempre ha querido y nunca ha podido. Son, simplemente, tan complicados o tan sencillos como las necesidades del usuario. Mas que unos programas de música, son toda una experiencia. Quién sabe, pueden ayudarte a descubrir un talento hasta este momento desconocido, dando rienda suelta a tu creatividad.

Kawasaki Rythm Rocker

Este programa (que realmente son dos en uno: uno de música y uno de gráficos) es una combinación impresionante de ritmos, efectos de percusión y batería, sonidos sintetizados, y efectos gráficos, disponibles al pulsar un botón para que des rienda suelta a tu imaginación.

El programa de música te permite:

1. Tocar el sintetizador con una serie de acompañamientos pregrabados de bajo.

2. Tocar el sintetizador con un acompañamiento de bajo y ritmo de tu propia creación.

3. Tocar efectos de batería y percusión con el bajo.

4. Grabar tus propias composiciones.

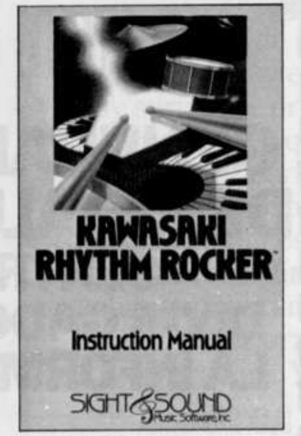
5. Tocar números de bajo y percusión creados con otro programa de Kawasaki "The Composer".

El programa de gráficos te permite:

1. Generar diseños gráficos multicolores incorporados en el software.

2. Modificar el tamaño, forma y ángulo de los gráficos.

3. Paralizar el dibujo en cualquier momento para hacer un scroll en pantalla.



4. Añadir otros efectos especiales de colores y tamaño a los gráficos.

Las dos filas superiores de teclas constituyen el teclado tipo piano, que quedan disponibles para tocar al cargar el programa. En este momento estamos en el modo sintetizador, y pulsando la tecla K podemos pasar a un efecto que "dobla" la nota, produciendo un curioso efecto gráfico a la vez si pulsamos la barra de espacio. Pulsando la tecla Shift se produce el vibrato, y la Shift Lock para sostenimiento y vibrato.

La tecla Return nos remite a una pantalla de ayuda donde se encuentra un resumen de todas las modalidades. Se vuelve a la pantalla principal pulsando la tecla S. Las tres modalidades son sintetizador (SYN), bajo (BASS) y Percusión (PERCS). La tecla F7 te permite cambiar de modo y un cuadrado pequeño a la derecha de la pantalla te indica el modo en el que te encuentras en cualquier momento.

En el modo de percusión el teclado desaparece y lo sustituye una "reja" en pantalla. En este modo cada tecla musical produce una percusión o un efecto sonoro diferente. Las teclas 3, 9, 0, —, HOME, y * abren la puerta a unos curiosos visitantes espaciales que aparecen en pantalla anunciando su visita con unos saludos de otras galaxias.

Pulsando la tecla F7 dos veces pasamos al modo del bajo que te permite tocar el bajo acompañado por el ritmo de fondo. Este modo (y el de sintetizador) también puede llevar los gráficos de "reja" si lo deseas. El programa lleva incorporado cuatro acompañamientos de bajo que puedes seleccionar usando la tecla F1, pero también puedes cargar otros acompañamientos de disco.

En la parte gráfica del programa todas las combinaciones están bien explicadas en el manual (disponible en castellano), pero haremos un pequeño resumen aquí. La tecla X sirve para empezar a dibujar y la tecla Z, para salir. La tecla D modifica el ángulo de la línea que estás dibujando, y la tecla F para elegir uno de cuatro tamaños. La tecla G varía la anchura de los dibujos y la tecla H borra cualquier dibujo previo y empieza uno nuevo que continúa durante un período de tiempo más largo. Este representa unas líneas más anchas y más largas. El propio programa también crea dibujos al azar y en cuanto termine uno empieza otro nuevo. Si practicas un poco, sabrás el momento más oportuno para parar un dibujo para que acompañe perfectamente el sonido, creando un número musical y gráfico usando tu propia imaginación. Cada vez que pulsas la tecla C, aparece una de las dieciséis combinaciones de color. La tecla M hace un scroll con el dibujo entero; la N para el scroll; la A te devuelve a la pantalla principal.

Existe otra pantalla gráfica (pulsando CTRL y Return en el modo del teclado) que se llama Gráfica de Sonido. El sonido se representa de una forma gráfica mediante unos cuadrados y rombos que aparecen de acuerdo con las notas tocadas y con el ritmo de la pieza.

El "Disk-Lounge" (Salón del Disco) es una modalidad que te permite trabajar directamente con la unidad de discos. Las funciones y las teclas que las controlan son las siguientes:

F1 — Unas versiones simplificadas de los comandos más utilizados (FORMAT, SCRATCH, RENAME, COPY, INITIALIZE, VALIDATE).

F3 — Para salvar tus composiciones o secuencias gráficas en disco.

F5 — Para acceder al directorio del disco.

F7 — Para cargar un fichero almacenado en el disco "Rythm Rocker" o cualquier disco tuyo.

Puedes grabar en la memoria del ordenador (tecla =) y volver a escuchar (tecla flecha a la izquierda) una melodía, o un acompañamiento parando cualquiera de las dos funciones con la tecla Commodore. Se pueden grabar hasta 256 notas. El color de borde de la pantalla cambia a rojo al llegar a 250 notas. Es importante observar esto porque, si grabas más de 256 notas, se considera como una nueva grabación y se borran todas las notas anteriores. Las teclas del cursor te permiten controlar la velocidad de la melodía o acompañamiento.

Existen dos funciones de doblaje para tocar o grabar el sintetizador o la percusión encima del bajo. Pulsando la tecla F3 se selecciona DUB-1 (para el sintetizador) o DUB-2 (para sonidos y efectos de percusión).

Este programa es un paquete musical y gráfico sumamente completo, pero, dada la claridad de las instrucciones en el manual, no supone ningún esfuerzo, incluso para una persona que no sepa nada de música ni de programación, proporciona un medio de creación sin límites para aquellos que ya disfrutan de unos conocimientos de estos terrenos.

3Kawasaki Synthesizer

Este paquete incluye dos discos "The Performer" (El Intérprete) y "The Composer" (El Compositor). El que vamos a considerar primero es el más sencillo de manejar: El Intérprete.

Dentro de este programa tenemos siete opciones que vamos a explicar a continuación. Todas están detalladas en la pantalla de ayuda (pulsando la tecla H).

1. Cómo tocar el teclado. Una serie de instrucciones en pantalla que explican la disposición de las teclas.

2. Escuchar melodías creadas por Ryo Kawasaki. Existen 13 canciones

"MUSIC PROCESSOR"

El lenguaje profesional

Con el lenguaje profesional 'MUSIC PROCESSOR' pueden jugar tus hijos y además aprender solfeo, composición, etc....
Estos son algunos de los comandos disponibles:

COMANDOS		INSTRUCCIONES	MODIFICACIONES	
APPEND	RECORD	(I) INSTRUMENTO Instrumento	W: NOTA REDONDA	•: STACCATO
AUTO	RENUMBER	(K) KEY Código	M: NOTA BLANCA	=: LEGATO
DELETE	SAVE	(L) LETRA Texto	Q: CUARTO DE NOTA	0, 1, 2, 3, 4, 5, 6, 7: OCTAVA
DIR	STEP	(!) PALABRAS Remarca sílabas	E: OCTAVO DE NOTA	
FIND	TRACE	(LE) LEGATO Voz en legato	S: DIECISEISAVO DE NOTA	
INSERT		(P) POKE Dirección, dato	T: TREINTA Y DOSAVO DE NOTA	
LIST		(PR) PRESET Preset para nuevo instrumento	.: NOTAS PUNTEADAS	
LOAD		(RHY) RHYTHM Construye ritmos	-: LIGADURA	
MERGE		(ST) STACCATO Pone voz en staccato	#: AGUDO	
PIANO		(V) VOLUME Selecciona volumen	F: BEMOL	
PLAY		(VI) VIBRATO Conecta el vibrato	N: NATURAL	

ARTICULO	VERSION	P.V.P.
-INCREIBLE MUSICAL KEYBOARD	DISC	13700
-INCREDIBLE MUSICAL KEYBOARD	CAS	12900
-MUSIC PROCESSOR	DISC	9900
-MUSIC PROCESSOR	CAS	7900
-KAWASAKI SYNTHESIZER	DISC	12200
-KAWASAKI SYNTHESIZER	CAS	9500
-KAWASAKI RHYTHM ROCKER	DISC	10700
-KAWASAKI RHYTHM ROCKER	CAS	7900
-3001 SOUND ODYSSEY	DISC	10500
-3001 SOUND ODYSSEY	CAS	7900

Transfórmese desde ahora en músico... creando, editando, componiendo y ejecutando sus propias piezas. El increíble teclado musical hace mucho más qe convertir su commodore en un sintetizador.

TODOS LOS PROGRAMAS EN CASSETTE O DISKETTE





KAWASAKI SYNTHESIZER
Transforme su Commodore-64 en un sintetizador musical programable. Dos diskettes o cassettes.

KAWASAKI RHYTHM ROCKER
Cree sonidos espaciales y gráficos multicolores al ritmo de la música electrónica.

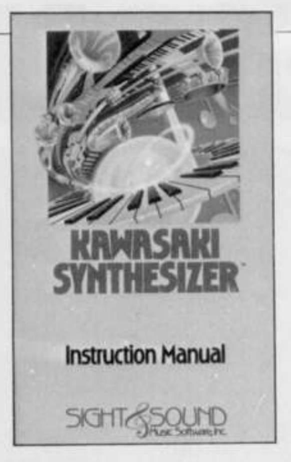
EXCEPCIONAL

3001 SOUND ODYSSEY
Explore los principios básicos de la música electrónica sintetizada hasta conseguir crear sus propios efectos de sonido.

casa de software s.a.

c/. aragón, 272, 8.°, 6.ª
tel. 215 69 52
08007 barcelona

DESEO RECIBIR INFORMACION:
Nombre y dirección: ______



pregrabadas acompañadas por una representación gráfica.

3. Este es el programa de demostración que te presenta a unos personajes curiosos.

4. Esta opción te permite seleccionar uno de 21 sonidos pregrabados.

5. Cómo modificar el rango de las notas del teclado o tocar en modo polifónico.

6. Cómo modificar las formas de onda:

F2 activa la triangular (piano, flauta). F4 activa la diente de sierra (trompeta, violín). F6 activa la onda pulsante (guitarra, clarinete). F8 activa el ruido blanco (batería, viento, ondas).

7. Esto es para tocar la Introducción y la Canción del Kawasaki Synthesizer.

Haciendo uso de todas las combinaciones posibles puedes llegar a producir hasta 500 sonidos diferentes.

Si "El Intérprete" sirve para introducirte en el mundo del sonido sintetizado (y de hecho, este programa es ideal como introducción a todos los programas de Ryo Kawasaki), "El Compositor" te deja mucho más margen para experimentar y explorar todas las posibilidades que se te ocurran. Las siguientes funciones del programa te permiten:

1. Tocar música utilizando una amplísima gama de sonidos y efectos especiales.

2. Grabar y volver a escuchar el bajo para poder usarlo como acompañamiento. (Existen ejemplos).

3. Grabar y reproducir composiciones de hasta tres voces. (También disponemos de ejemplos).

4. Crear sonidos y efectos especiales y almacenarlos en disco.

La pantalla del teclado te permite jugar con las siguientes posibilidades:

F1, F3, F5, F7: elegir entre cuatro octavas: alta, mediana, baja y superbaja.

F2, F4, F6, F8: elegir entre la onda triangular, onda diente de sierra, onda pulsante, ruido blanco.

F: activar efecto Wah-Wah.

G: desactivar este efecto.

Barra de espacio: Pantalla para editar sonidos.

—V—: Cargar el sonido.

CRSR —V: Monofónico.

CRSR a la derecha: Polifónico.

Tecla de dos puntos (:): Activa el vibrato y el punto y coma (;) lo desactiva.

La pantalla de editar sonidos te permite realizar una serie de funciones importantes:

1. Editar sonidos individuales.

2. Salvar y cargar sonidos prefijados en discos.

3. Grabar, editar, y reproducir secuencias (composiciones).

4. Salvar y cargar las secuencias en disco.

Desde la pantalla de editar sonidos se puede acceder al Modo de Mezclar pulsando la tecla M. Esta modalidad te permite elegir una forma de onda diferente para cada una de las tres voces.

Hemos resumido brevemente las funciones principales del "Kawasaki Synthesizer" para dar una idea de la versatilidad de estos programas. Las instrucciones del manual (traducido al castellano) son fáciles de seguir, lo que te permite ponerte a tocar, jugar, experimentar, componer, aprender y en general, pasártelo muy bien nada más cargado el programa.

3001 Sound Odyssey

En mi opinión nos encontramos frente a uno de los mejores programas musicales del mercado; trataré de resumir las cualidades que me han llevado a esta conclusión.

El programa viene acompañado de un manual detallado y correctamente traducido al castellano. Esto facilitará el manejo del programa y el descubrimiento de sus posibilidades.

Al cargar el programa no encontramos frente a la presentación hasta que pulsemos una tecla, en ese momento pasaremos al menú principal, en el que podemos ver las opciones del sintetizador, el tutorial sobre síntesis de sonido, la demostración y la opción de la presentación. Para seleccionar la opción podemos utilizar el joystick o el teclado (tanto en ésta como en todas las demás opciones del programa).

Si para empezar vemos la demostración, nos dejará con la boca abierta y nos dará ánimos para continuar el aprendizaje del manejo del programa. En la demostración visualiza en pantalla los ajustes que realiza sobre los distintos mandos del sintetizador mientras escuchamos los efectos producidos sobre el sonido.

Posteriormente podemos pasar al tutorial, en él podemos aprender todo aquello que necesitemos saber sobre la síntesis de sonido. En el menú del tutorial encontramos una lista de nueve lecciones, en cada una de ellas tendremos tres opciones: una informativa (Show me) consistente en explicaciones (texto), otra didáctica con demostración (visual y sonora) y la tercera tomar el control sobre esa parte del sintetizador para experimentar por nosotros mismos (Try me).

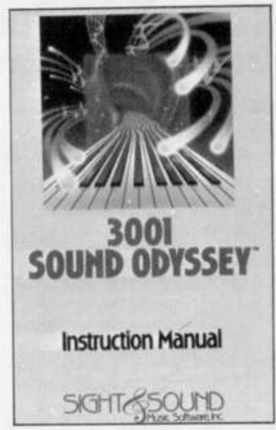
Los temas estudiados son: introducción, volumen, pitch, ADSR, formas de onda, filtros, modulación en anillo y sincronización, modulación con LFO, seguidor de envolvente.

El único problema es que el texto aparece en inglés, pero los ejemplos que incorpora son tantos y tan claros que en realidad no importa mucho. Quizás la casa distribuidora lo traduzca en el futuro, pero no parecen tener intenciones inmediatas de hacerlo, aunque el manual ya está traducido.

Las otras opciones que nos quedan son la de la presentación, que la hemos visto al cargar el programa, y la más importante, que es la utilización práctica del sintetizador. Cuando seleccionamos esta opción pasamos a encontrarnos frente al tablero de un sintetizador con un teclado (tipo piano) de cuatro octavas completas en el que vemos la nota pulsada; los controles del sintetizador son:

Volumen, y para cada una de las voces interruptores, ADSR, forma de onda, anchura de impulsos, modulación en anillo, sincronización, ajuste (pitch), filtros elimina-banda, pasabanda, pasa-altos, pasa-banda, control de resonancia, corte, velocidad del LFO, sus filtros, seguidor de envolvente, etc.

Para ejecutar una melodía disponemos de las dos filas superiores del teclado, la superior son las teclas negras y la inferior las blancas, distribuidas igual que en un piano. Si disponemos también del teclado de la misma casa



podemos instalarlo, con lo que dispondremos de un auténtico teclado de piano. Para cambiar las octavas sobre las que estamos tocando, pulsaremos las teclas commodore y shift (izquierdo).

También se puede acompañar cada nota con cambios de color en el borde de la pantalla; para ello basta pulsar la C (Color). Visualizaremos la nota pulsada en la notación de letra-número de octava si pulsamos la tecla N (Nota). Estos controles actúan a modo de interruptores ON/OFF.

Todos los mandos del sintetizador los

podemos regular con la ayuda del joystick o las teclas de cursor y la de flecha a la izquierda, con esto el sintetizador se puede utilizar en el modo directo mientras ejecutamos una melodía en el teclado, pero como la mayoría de los efectos que posiblemente desearemos obtener serán complicados, es necesario pasar al estudio de las posibilidades de Preset y Secuenciador.

Disponemos de un máximo de noventa y nueve presets o preajustes programables, además de los ocho instantáneos asignados a las teclas de función. Cada uno de estos preajustes puede ser borrado, modificado, salvado en disco o cargado en éste, por lo que podremos trabajar para conseguir la variedad de sonidos que deseemos y guardar nuestro trabajo en disco para utilizarlo posteriormente. Para obtener los nombres de los trabajos ya existentes en el disco pasaremos a la opción LOAD y teclearemos DIR (directorio).

También disponemos de un secuenciador con el que podemos memorizar, ejecutar, salvar o cargar melodías desde el teclado. El secuenciador tiene un control de ritmo R, con un margen de ajuste desde +99 hasta -99, lo que permite memorizar a 'cámara lenta' las secuencias de ejecución más complicadas y después reproducirlas a velocidad rápida o normal, consiguiendo con ayuda de este control unos efectos impresionantes (imaginar la sensación de habilidad que puede causar a los oyentes).

En el disco del programa podemos encontrar un par de ejemplos de utilización del secuenciador (demo1 y demo2), y cinco de preajustes (presets, ballgame, canon, circus y boogie). Los cuatro últimos permiten demostrar un curioso modo de utilizar el sintetizador; dejando una nota pulsada y cambiando los ajustes consigue efectos maravillosos (probar boogie...).

Creo que es difícil contar las posibilidades de este programa en una reseña, y es mejor ver el disco de demostración en uno de los distribuidores.

Music processor

Como programador creo que éste es el paquete que más me ha gustado, ya que dispone de un nuevo lenguaje de programación totalmente destinado a la música y con extensiones que permiten incorporar la letra a la música, manejar los colores, etc.

En la primera sección del manual te indica que existen dos modos de aprender a utilizar el programa, el primero de ellos es el rápido o experimental, consistente en cargarlo, pulsar F7 para obtener un escueto resumen de sus funciones y lanzarse a probar todo, el segundo de ellos es más lento pero seguro, consiste en seguir paso a paso las secciones del manual (en castellano).



En el menú principal del music processor tenemos una serie de títulos que podemos seleccionar para escuchar y ver ejemplos musicales; para ello pulsamos cualquier tecla de la A a la N. Para pasar al modo de piano pulsaremos la tecla F1, y nos encontraremos frente a un pentagrama doble con las claves de sol y fa, en él veremos cada nota que pulsemos. Para seleccionar instrumentos (entre los 99 disponibles), basta pulsar las teclas F1 o F3, y para cambiar la octava sobre la que estamos tocando pulsamos F5 ó F7. El teclado del ordenador está dividido, las dos filas superiores son las teclas de la octava más alta, y las inferiores las de la octava más baja. Si queremos accionar el acompañamiento, pulsaremos la tecla de flecha a la izquierda.

Si desde el menú principal pulsamos F3, estaremos escuchando una caja de música que interpretará todas las piezas del disco (una detrás de otra).

Con F2 pasamos al modo de grabación, toda la melodía se memorizará conforme tecleamos. Más tarde podremos reproducirla pulsando F4 desde el menú principal. Si en el modo de grabación pulsamos la tecla RETURN, aparecerá el cursor en el ángulo inferior izquierdo, ahora podemos teclear comandos de los que aparecen en el manual, por ejemplo podemos hacer que aparezca la letra mientras suena la melodía.

Con la tecla F6 hacemos que se repita una melodía, y con F8 pasamos al modo RANDOM (aleatorio), en el que el computador toma la iniciativa y se dedica a producir sus propias composiciones. Este modo de funcionamiento es útil para indagar la variedad de sonidos prefijados (moviendo el joystick o con las teclas de cursor) mientras el ordenador toca sus 'obras maestras aleatorias'.

Por último, con F5 pasamos al modo de edición (el más interesante para los programadores), es el modo de funcionamiento más potente del Music Processor, ya que podemos utilizar el editor como si estuviéramos trabajando con un programa en Basic, almacenando las líneas del 'programa' en la memoria RAM, salvándolas con SAVE, cargándolas con LOAD.

Para añadir líneas teclearemos un número de línea entre 1 y 65534 seguido de los comandos que deseemos ejecutar. Para borrarlas tecleamos sólo el número de línea y pulsamos RETURN (igual que en Basic). Mientras se ejecuta una melodía podemos cambiar el ritmo pulsando cualquiera de las teclas del cero al nueve. También podemos pasar al modo de TRACE pulsando la tecla T, o al de paso a paso pulsando la S (STEP), para volver al normal pulsaremos la P (PLAY). Si queremos detener la interpretación de una melodía pulsamos la Q o STOP, y pasaremos al modo normal de edición del music proccesor.

Disponemos de una gran variedad de comandos, por ejemplo: Append (para mezclar melodías), Auto (para numerar automáticamente las líneas), Bye (para salir a Basic), Delete (para borrar parte de un programa), Dir (directorio del disco), Disk (envía comandos al disco), Find (busca algo en un programa), Fast Forward (reproduce la melodía a gran velocidad), Free (indica la memoria libre), Insert (inserta líneas en medio del programa), Jukebox (permite ejecutar melodías de la caja de música), Keyboard (permite cambiar la configuración del teclado), List (lista el programa-melodía), Load (lo carga), Menu (vuelve al menú principal), Merge (mezcla tres melodías en una sola), New (borra la melodía-programa), Piano (configura el teclado como un piano), Play (interpreta una melodía), Plist (lista a la impresora), Presets (permite definir los instrumentos seleccionables para una melodía —10 entre los 99—), Random (activa el modo aleatorio), Record (pasa al modo de piano mientras memoriza lo que se teclea, Renumber (renumera las líneas), Repeat (igual que play, pero continúa tocando hasta que pulsemos Q), Save (guarda en disco), Step (permite ejecutar paso a paso la melodía), Trace (interpreta la melodía mientras visualiza cada línea de comandos), Key (cambia la clave de la melodía), Lyrics (permite visualizar la letra) (shift 1) (resalta una parte de la letra), Legato, link, measure, etc.

Comprenderéis que no cuente todas las posibilidades de este programa ni todos los comandos que podemos utilizar, ya que tiene de todo tipo (para el ritmo, ADSR, filtros, etc.), la lista sería largísima, pues el manual es bastante escueto y tiene más de sesenta páginas dedicadas a estos comandos de programación.

Lo más interesante es saber que disponemos de un lenguaje de programación musical con el que podemos hacer mil y una maravillas sin quebrarnos la cabeza con cantidad de pokes raros y cosas de ésas, ya que los comandos de control de la música se llaman por su nombre musical, por ejemplo "Vibrato,

EN ALEMANIA, U.S.A., INGLATERRA, HOLANDA, FRANCIA, CANADA, Y AHORA EN ¡ESPAÑA!

A PARTIR DE AHORA LOS ENCONTRARA EN: LIBRERIAS QUIOSCOS Y TIENDAS DE INFORMATICA





IDEAS PARA EL COMMODORE 64

Casi todo lo que se puede hacer con el Commodore 64, está descrito detalladamente en este libro. Su lectura no es tan sólo tan apasionante como la de una novela, sino que contiene, además de listados de útiles programas, sobre todo muchas, muchas aplicaciones realizables en el C64. Se ha valorado especialmente, que el libro sea de fácil comprensión pa los no iniciados. Un extracto del temario: El ordenador escribe poesías, tarjetas de invitación, cartas publicitarias personalizadas, coste por kilómetro de su coche, cálculo de costes de construcción, calculadora de bolsillo, fichero de recetas, inventario, fichero personal de la salud, plan electrónico de dieta, diccionario inteligente, CAD para trabajos manuales, optimización de rutas, escaparates publicitarios, juegos de estrategia. En parte hay listados de programas listos para ser tecleados, siempre que ha sido posible condensar «recetas» en una o dos páginas. Si hasta el momento no sabía que hacer con su Commodore 6 ¡después de leer este libro lo sabrá seguro!
EL LIBRO DE IDEAS DEL COMMODORE 64, 1984, más de 200 páginas, ptas. 1.600,-

64 CONSEJOS Y TRUCOS

CONSEJOS Y TRUCOS, con más de 70.000 ejemplares vendidos en Alemania, es uno de los libros más vendidos de DATA BECKER. Es una colección muy interesante de ideas para la programación del Commodore 64, de POKEs y útiles rutinas e interesantes programas. Del contenido: Gráficas 3D en Basic - gráficas de barras en colores - definición de un juego propio de caracteres - simulación del ratón con el joystick - Basic para avanzados - el C64 habla castellano - CP/M en el Commodore 64 - conexión de impresoras a través del port de usuario - transmisión de datos desde y hacia otros ordenadores - sintetizador en estéreo - recuperar un fichero, que no ha sido cerrado correctamente - generar una línea Basic en Basic - el buffer del cassette como memoria de datos - multitasking en el Commodore 64 - la página cero - GOTO, GOSUB y RESTORE con números de línea calculados - función INSTR y STRING - repetición automática de todas las teclas. Todos los programas en lenguaje máquina con programas cargadores en Basic.
64 CONSEJOS Y TRUCOS, 1984, 364 pág. P.V.P. 2.800,- ptas.

PEEKS Y POKES PARA EL COMMODORE 64

Con importantes comandos PEEK y POKE se pueden hacer también desde el Basic muchas cosas, para las que se necesitarían normalmente complejas rutinas en lenguaje máquina. Este libro explica de manera sencilla el manejo de PEEKs y POKEs. Con una enorme cantidad de POKEs importantes y su posible aplicación. Para ello se explica perfectamente la estructura del Commodore 64: Sistema operativo, interpretador, página cero, apuntadores y stacks, generador de caracteres, registros de sprites, programación de interfaces, desactivación del interrupt. Además una introducción al lenguaje máquina. Muchos programas ejemplo.
177 pág. P.V.P. 1.600,- ptas.

TODO SOBRE EL CASSETTE PARA EL C-64 Y VIC-20

Un excelente libro, que le mostrará todas las posibilidades que le ofrece su grabadora de casettes. Describe detalladamente, y de forma comprensible, todo sobre el Datassette y la grabación en cassette. Con verdaderos programas fuera de serie: Autostart, Catálogo (¡busca y carga automáticamente!), backup de y a disco, SAVE de áreas de memoria, y lo más sorprendente: un nuevo sistema operativo de cassette con el 10-20 veces más rápido FastTape. Además otras indicaciones y programas de utilidad (ajuste de cabezales, altavoz de control). **190 pág. P.V.P. 1.600,- ptas.**

MSX

El libro contiene una amplia colección de importantes programas que abarcan, desde un desensamblador hasta un programa de clasificaciones deportivas.
Juegos superemocionantes y aplicaciones completas. Los programas muestran además importantes consejos y trucos para la programación.
Estos programas funcionan en todos los ordenadores MSX, así como en el SPECTROVIDEO 318 328.
ESTRACTO DEL CONTENIDO:
Volcado memoria hexadecimal. Editor gráficos. Editor de sonido.
Escritura de ordenador. Lista referencia de variables.
Calendario. Desensamblador.
ADMINISTRACION de una colección de discos L.P.
HOLLOW - JUEGO DE LAS CEREZAS.
DIAGRAMAS DE BARRAS.
TABLAS DEPORTIVAS.
194 pág. 1.985 P.V.P. 2.200,- ptas.

¡TODOS LOS PROGRAMAS DE CADA LIBRO ESTAN A SU DISPOSICION EN DISCO O EN CASETTE!

FERRE - MORET S.A.

TUSET, 8 ENTLO. 2.° – ☎ 218 42 04 - 218 40 58
TELEX 97851 CBCT E – 08006 - BARCELONA

ESTAREMOS EN INFORMAT 85 BARCELONA

BOLETIN DE PEDIDO

FERRE - MORET S.A. Tuset n.° 8, entlo. 2.° Tel. 218 02 93
BARCELONA 08006

Deseo adquirir ______
Gastos envío: 300 ptas. □ Adjunto cheque □ Reembolso más gastos del mismo
NOMBRE ______
DIRECCION ______

voz, velocidad, profundidad" activa el vibrato en la voz..., con una velocidad..., y con una profundidad de modulación..., "Volumen N" pone el volumen a N, etc. Supongo que a muchos les agradará disponer de estos comandos para olvidarse de los pokes tan difícil de recordar.

Si encuentras que el teclado del C-64 te resulta incómodo dado la disposición de las teclas, la misma casa vende un teclado tipo piano, del tamaño del ordenador que encaja encima de éste dejando un hueco para poder acceder a las teclas de función, de forma que puedes tocar tranquilamente sin confundirte. El programa que acompaña el teclado es una demostración muy completa de todos los programas de Ryo Kawasaki. Adicionalmente, este programa te permite tocar el teclado con el sonido de piano, sintetizador o bajo.

Artículo	Versión	P.V.P.
Increíble musical Keyboard	Disc	13.700
Increíble musical Keyboard	Cas	12.000
Music processor	Disc	9.900
Music processor	Cas	7.900
Kawasaki Synthesizer	Disc	12.200
Kawasaki Synthesizer	Cas	9.500
Kawasaki Rythm Rocker	Disc	10.700
Kawasaki Rythm Rocker	Cas	7.900
3001 Sound Odyssey	Disc	10.500
3001 Sound Odyssey	Cas	7.900

Estos programas son comercializados por la empresa Casa de Software S.A.
c/ Aragón, 272, 8, 6.
08007 Barcelona
Tel.: (93) 215 69 52

Music Master

El Music Master es un programa destinado a facilitar el descubrimiento de las posibilidades del Commodore 64 para producir sonidos, permite variar los parámetros que controlan la forma de onda, envolvente y filtros que, una vez fijados para cada una de las tres voces, hacen posible tocar una melodía desde el teclado como si se tratase de un piano. Por tanto, este programa funciona normalmente como un sintetizador programable monofónico, aunque con un poco de tiempo se pueden llegar a obtener acordes.

El Music Master está dotado de un generador de ritmos automático que permite generar música de fondo o acompañamiento. Este generador de ritmos tiene un total de treinta y cuatro diferentes, y el usuario puede definir otros dos más.

Además de utilizarlo como si fuese un piano de acompañamiento, puede utilizarse en el modo programado. En este modo puede editarse una melodía nota a nota o al normal de ejecución, para



posteriormente reproducirla a la misma o diferente velocidad. Podemos programar y grabar hasta tres partituras (una para cada generador), creando una melodía a tres voces.

Cuando editamos una canción disponemos de memoria para almacenar un máximo de mil quinientas notas para cada voz.

En la pantalla del ordenador podemos ver una sección del teclado de un piano representado o pentagramas, de modo que parezca algo normal para aquellos que no estén acostumbrados a la representación de notas en forma de letras. El manual dice que los músicos no tendrán dificultades para adaptarse a esta notación musical aunque nunca la hallan utilizado antes, y los profanos se adaptarán pronto al pentagrama y al teclado de piano, pasando posteriormente al sistema de composición a tres voces con letras.

El Music Master permite también crear efectos especiales con el chip de sonido del Commodore 64 (SID), entre ellos el wah-wah, vibrato, reverberación y tremolo, utilizando el tercer oscilador para modular otros registros del SID. Estos últimos efectos requieren algún tiempo para poder manejarlos, y no están destinados a los principiantes, pero pueden intentarse cuando tengamos alguna práctica en el manejo de este programa.

Una característica muy importante y peculiar del Music Master es que incorpora un generador de programas Basic, que nos proporciona un modo de cubrir la falta de comandos de música en el Basic del 64.

Con este programa puedes generar una subrutina musical que podrás añadir a tus propios programas, para utilizarlo sólo tienes que editar la música (simplemente utilizando los comandos normales del Music Master), y entrar en la opción del generador de programas Basic pulsando las teclas Shift y W. En este momento el programa te pregunta el número de línea inicial y el incremento entre líneas, le daremos dos valores que permitan colocar la numeración de líneas donde nos venga mejor en el rango de 0 a 63.9999, y luego hacer el SAVE.

Para utilizar la rutina musical que hemos generado, debemos hacer un MERGE con alguna de las extensiones Basic disponibles en el mercado (Simon's, Breden's, Vic-tree, etc.) o alguna rutina de las aparecidas en la sección de Magia. Otro modo es cargar primero la rutina musical y luego teclear el resto del programa.

Debemos hacer una aclaración, y es que los efectos especiales y de acompañamiento no son pasados al programa generado en Basic, ya que todas estas maravillas se suelen hacer un código máquina (podéis verlo en este mismo número en el artículo Serenata SID).

Como evaluación general, podemos decir que el programa está bastante bien, pese a no llegar al nivel de posibilidades sonoras de un Kawasaki Synthesizer o un Music Processor, pero, por otro lado, la opción de poder generar rutinas en basic para incorporarlas a nuestros propios programas no la incluyen otros programas de mayor calidad musical. Demos considerar que éste es un programa para aquellos que quieren iniciarse en los aspectos musicales de su ordenador sin gastar mucho dinero, para más adelante subir de nivel con otro programa más completo (en caso de que le guste ese campo), aunque también lo podremos considerar como un utilitario para facilitar la edición de las rutinas musicales de nuestros juegos.

Este programa lo comercializa en España la casa Sakati de Madrid y sus señas son:
Sakati
Ardemans, 24
28028 Madrid
Tel.: (91) 256 77 94
El precio del Music Master en cinta es de 7.000 ptas. y la versión en disco 7.500 ptas.

Music 64

MUSIC 64 es un sintetizador monofónico o polifónico que consiste en un teclado conectado a un C-64. Este ordenador dispone de unos circuitos interiores que le permiten funcionar como un sintetizador. Dispone de tres osciladores con forma de onda que van desde la triangular, a la cuadrada, diente de sierra, impulsos y ruido; la unidad de control ADSR, una unidad filtro y la posi-



bilidad obvia de controlar los osciladores en amplitud y frecuencia. Todo esto era difícilmente accesible para una persona que no fuera un programador muy experto, pero en este paquete también disponemos de un teclado.

El instrumento es bastante fácil de hacer funcionar porque basta con conectar el teclado al cable suministrado, y este último al conector trasero del Commodore. El programa viene en disco o cassette.

Esencialmente el uso del MUSIC 64 se divide en dos fases diferentes: la definición de los instrumentos y la ejecución de música en tiempo real. En la definición de instrumentos, el sistema presenta una pantalla que simula el panel de un sintetizador con la posibilidad de cambiar todos los parámetros visualizados. Existen unos instrumentos prefijados en las dos maneras de funcionamiento, pero como disponemos de la facilidad de probar los resultados inmediatamente, este instrumento es idóneo para experimentar con todos aquellos efectos que se pueden hacer en la música hoy en día. En el modo de ejecución basta pulsar una tecla del ordenador para que se reproduzca en seguida lo que nuestros dedos han tocado. Si en este momento queremos cambiar de instrumento, basta con pulsar la tecla correcta para volver a la situación anterior, preparados para crear nuevas melodías. Llegados a este momento, tenemos que plantearnos una pregunta: ¿Si en el interior del C-64 existen solamente tres osciladores, mientras que en MUSIC 64 en el modo polifónico se pueden obtener ocho, de dónde han salido las otras voces? Milagros de la programación.

El equipo necesario se compone del teclado polifónico, unidad de disco o cassette, interface, caja de ritmos y el programa "Organo y Ritmos".

El programa funciona en dos modalidades: de editar y de tocar. El modo de editar está compuesto por dos partes, la de ritmo y la de voz, en las que se pueden crear o modificar los ritmos y parámetros de los efectos pregrabados. El modo de tocar permite la utilización de los efectos creados con la posibilidad de modificar la selección del ritmo o de las voces en tiempo real.

En el modo de editar ritmos se dispone de las siguientes funciones:

—Crear un fichero de 9 ritmos diferentes.

—Modificar los ritmos.

—Dar un nombre a cualquier ritmo creado.

—Salvar un ritmo en disco.

—Cargar un ritmo de disco.

—Escuchar los ritmos mientras se realizan las modificaciones.

Cómo crear o modificar un ritmo. El diseño rítmico incluye una extensión de 32 asientos rítmicos de 7 instrumentos.

Los siguientes parámetros sonoros también se pueden modificar:

1.—Filtro.
2.—Resonancia.
3.—Ataque.
4.—Decaimiento.
5.—Sostenimiento.
6.—Relajación.
7.—Volumen.
8.—Transposición.
9.—Forma de onda.

El precio del teclado y el software es de 37.500. La empresa Albareda nos comunica que la caja de ritmos está recién llegada y el precio de comercialización andará por las 19.000 pesetas.

La casa que comercializa estos productos es:

ALBAREDA
c/ Carmen, 19
Tárrega.
Lérida
Tel.: (973) 31 04 02. 31 23 51

Commodore 128

Promete mucho, muchísimo. Estamos deseando verlo. Si efectivamente dispone de todo lo que la casa Commodore ha anunciado, nos espera un artículo de auténtico lujo. De momento lo único que podemos hacer es enseñaros su cara bonita y puntualizar un par de datos técnicos.



—128K con ampliación hasta 512K

—Un potente lenguaje de programación que permite el uso de más de 100 sentencias y comandos.

—Pantalla de 25 líneas de 80 columnas y 25×40 para compatibilizarlo con el C-64.

—Microprocesador 8502 y Z80 4MHz.

—El teclado principal tiene la misma disposición que el del C-64, además de un teclado numérico.

—Sistema operativo CP/M y Kernal.

Quizás su punto más fuerte será su compatibilidad con el C-64 y sus periféricos. ¡A ver si es verdad!

El PC de Commodore



La segunda promesa para estos meses es la presentación del nuevo ordenador de la gama profesional de Commodore; dicen que es totalmente compatible con el PC de la casa IBM; si esto es cierto y lo vemos, pronto será un bombazo que hará temblar muchas marcas de ordenadores.

De momento sólo sabemos el precio de lanzamiento, unos mil quinientos dólares, y ha sido diseñado en Alemania.

Si es compatible en un cien por cien con el PC de IBM, tendrá a su alcance más de diez mil aplicaciones profesionales, todos los lenguajes de programación imaginables (APL, Pascal, Modula-2, C, Basic, Cobol, etc.), y una gran variedad de utilitarios para el desarrollo de programas profesionales.

¡Y... a ver cuándo lo vemos por estos lares!

ESTE ES EL SIMBOLO DE COMMODORE, COMPAÑIA AMERICANA, LIDER MUNDIAL EN NUMERO DE ORDENADORES INSTALADOS.

Este PC de Commodore ha hecho dudar a más de uno.

La oferta del mercado de los ordenadores PC dejaba hasta hoy muy sencilla la elección. Sin embargo, Commodore, líder reconocido en varios sectores de la informática, ha ofrecido una respuesta alternativa que atiende plenamente las exigencias empresariales y de profesionales liberales: su nuevo ordenador PC.

El nuevo Commodore PC dispone de una versatilidad acorde con una tecnología depurada en constante evolución y compatible con el software standard que más le suena.

commodore PC

Estos avances, y un precio realmente interesante, han planteado serias dudas entre los profesionales más cualificados a la hora de elegir un buen PC.

Sin duda Commodore, con el mayor número de ordenadores vendidos en el mundo se afianza en el campo empresarial con mucha fuerza.

Si está interesado en conocer más de cerca el nuevo PC de Commodore, pregunte en cualquier concesionario Commodore, le sacará de dudas.

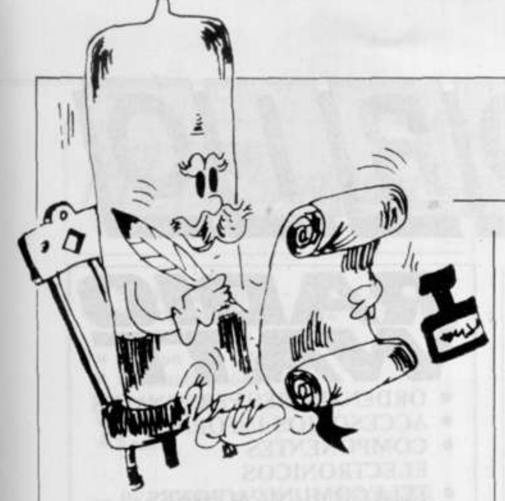
PRINCIPALES CARACTERISTICAS

– 256 K de RAM de 9 bits – Zócalos para 512 K más – 2 unidades de diskete de 360 K – Disco 10 Mb opcional – Interfases serie y pararelo, incluidos – 5 slots compatibles – Alta resolución incluida – El mejor precio en esta categoría.

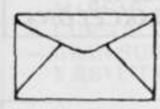


Microelectrónica y Control *c/ Valencia, 49-53 08015 Barcelona - c/ Princesa, 47 3.º G 28008 Madrid*
Unico representante de Commodore en España.

CARTA BLANCA



UN JOYSTICK COMO RATON

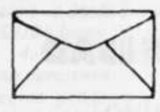


Quiero felicitaros antes que nada por el trabajo que estáis llevando a cabo. El nivel que estáis consiguiendo en la revista es bastante bueno a mi juicio, tanto para los que empiezan como para los que ya saben cosas.

La sugerencia que os mando es para los que tenemos joystick. Nos vendría muy bien alguna rutina en código máquina que nos permitiera utilizar el cursor sin necesidad de las teclas. En programación Basic es fácil hacerlo, pero lo que propongo es poder utilizarlo a modo de ratón para facilitar el manejo de la pantalla. ¿Es mucho pedir? A ver si alguien se anima.

Manuel del Arbol Sánchez
C/Arrumbadores, 4 - 4
Montilla (Córdoba)

SOBRE EL LENGUAJE MAQUINA



1. En el número 8 de octubre, en la página 64 se hace una demostración de cómo traducir un programa en lenguaje ensamblador a código L.M. por el sistema manual. ¿Podéis aclarar un poco más cómo se calcula el valor $FA del operando de la instrucción "BNE LAZO" en la dirección de memoria $0344?

2. ¿Qué significa el símbolo "#" (chr$35) generalmente colocado entre la instrucción y su operando, en el lenguaje ensamblador?

3. ¿Qué es una "página de memoria"? (queda bien entendido el concepto de "página cero", pero al parecer hay más páginas).

4. Al hacer el desensamblado con el programa "Periscope" de la memoria del Vic-20 en su configuración base, por ejemplo, de la zona dedicada a la ampliación de 8K o más, aparecen en la misma instrucciones de L.M. ¿A qué es debido? ya que al parecer si el Vic no está ampliado estas zonas deberían estar vacías.

5. (y última) Cuando desensamblamos un programa con el "Periscope" y aparece la "instrucción" ???, ¿Qué significa?

Francesc Andreu Bertran
C/Roca, 2. 2-4
Sabadell

1.—Cuando el salto es relativo, quiere decir que el ordenador va a saltar X posiciones hacia adelante o hacia atrás. Si el salto es hacia adelante, el siguiente byte a la instrucción se cuenta como el 00, el siguiente como el 01, etc. Si haces BNE $00, el programa sigue como si no hubiera salto (porque pasa a la siguiente instrucción). Cuando el salto es hacia atrás, comienzas a contar desde 00 —siguiente posición a la instrucción— hasta la posición a la que se va a realizar el salto, en este caso sería lo siguiente:

POS 0346 = 00
POS 0345 = FF
POS 0344 = FE
....
POS 0340 = FA (posición a la que se realiza el salto).

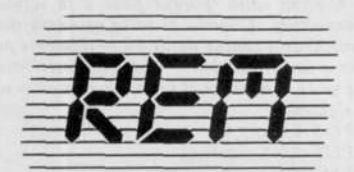
Si utilizas un ensamblador (no el "Periscope") no hace falta que calcules el valor, sino que le introduces la posición a la que se salta y él mismo se encarga de hacer los cálculos.

2.—El símbolo "#" quiere decir que el modo de direccionamiento es inmediato. Si haces LDA # $C5, cargará el acumulador con $C5, pero si haces LDA $C5, cargará el acumulador con lo que contenga la posición $C5 (en decimal 197) —Este segundo es el direccionamiento en página cero.

Ambos modos de direccionamiento ocupan un solo BYTE (códigos $A9 y $A5, respectivamente), y el símbolo "#" es meramente para diferenciarlos a la hora de leerlos en un listado.

3.—Efectivamente. El Vic (y también el 64) tienen 256 páginas —de la 0 a la 255—. Una ►

DIRECTORIO



• Ordenadores personales Hard y Soft.
• Cursos de Basic.

RENOVACION EN MARCHA, S.A.

OFICINAS
C/. Espronceda, 34-2º int.
28003 MADRID
Teléfono **(91) 441 24 78**

REM SHOP 1
C/. Galileo, 4 - 28015 MADRID
Teléfono **(91) 445 28 08**

REM SHOP 2
C/. Dr. Castelo, 14 - 28009 MADRID
Teléfono **(91) 274 98 43**

REMSHOP-3
C/. Modesto Lafuente, 33
28003 MADRID
Teléfono **(91) 233 83 19**

REM SHOP - BARCELONA
C/. Pelayo, 12 - Entresuelo J
Teléfono **(93) 301 47 00**

REM SHOP - LAS PALMAS
Gral. Mas de Gaminde, 45
Teléfono **(928) 23 02 90**
(Inauguración) 25/2/85

casa de software s.a.

c/ aragón, 272, 8º, 6.ª
tel. 215 69 52 - 08007 barcelona

- **Software profesional para C-64**
- **Con distribución productos DIGITAL RESEARCH**

SOFTWARE ESPAÑA

Avda. de Arteijo, 19
14004 La Coruña
Teléf. 25 51 72

Especializados en software para Commodore-64 Spectrum y MSX

OPORTUNIDAD

Por cambio de negocio se venden muebles de oficina en lote completo o por separado.

Todo el mobiliario de diez despachos, sala de juntas, etcétera.

Teléfs. 231 25 17 - 231 24 62
Preguntar por Angel Rodríguez.



HACEMOS FACIL LA INFORMATICA

- SINCLAIR • SPECTRAVIDEO
- COMMODORE • DRAGON
- AMSTRAD • APPLE
- SPERRY UNIVAC

Modesto Lafuente, 63 Telf. 253 94 54 28003 MADRID	Colombia, 39-41 Telf. 458 61 71 28016 MADRID
José Ortega y Gasset, 21 Telf. 411 28 50 28006 MADRID	Padre Damián, 18 Telf. 259 86 13 28036 MADRID
Fuencarral, 100 Telf. 221 23 62 28004 MADRID	Avda. Gaudí, 15 Telf. 256 19 14 08015 BARCELONA
Ezequiel González, 28 Telf. 43 68 65 40002 SEGOVIA	Stuart, 7 Telf. 891 70 36 ARANJUEZ (Madrid)

ELECTROAFICION COMPUTER

C/Villarroel, 104
08011 Barcelona - Tel. 253 76 00 - 09
C/Gran Vía Corts Catalanes, 559
08011 Barcelona - Tel. 254 23 19

- MICROORDENADORES
- ACCESORIOS INFORMATICOS
- SOFTWARE
- RADIOAFICIONADOS



- ORDENADORES PERSONALES
- ACCESORIOS INFORMATICA
- COMPONENTES ELECTRONICOS
- TELECOMUNICACIONES

Paseo de Gracia 126-130
Tel. 237 11 82*. 08008 BARCELONA

CENTRO DE INFORMATICA

Las Rozas - Majadahonda
EMPEZAMOS
Cursillos en BASIC cada 15 días
Tenemos todo lo referente al COMMODORE 64
Teléfono: 637 31 51



DISTRIBUIDORES DE:

ORIC-1
CASIO FP200
ROCKWELL-AIM-65
VIDEO GENIE-EG-2000
CASIO FX-9000P
SINCLAIR SPECTRUM
OSBORNE 1
DRAGON-32
NEW BRAIN
EPSON HX-20

C/SANDOVAL, 3, 4, 6 MADRID 28010
TELEFONOS 445 18 33/18 70
TELEX 47784 SAVL-E

VENTAMATIC

- VEN A CONOCERNOS. Somos los SUPER-ESPECIALISTAS del COMMODORE 64 y lo tenemos TODO para tu COMMODORE 64 (incluyendo lo último en accesorios y programas musicales y MIDI). SOLICITA CATALOGO COMPLETO.
- VENTAMATIC - c/. Córcega, 89, entlo. 08029 BARCELONA - Tel.: (93) 230 97 90. Metro ENTENZA Línea V. Bus: 41, 27, 15, 54, 66.

Departamento de Envíos y Venta por Correo:
- VENTAMATIC - Avda. de Rhode, 253 - ROSES (Girona) - Tel.: (972) 25 79 20.



página es un grupo de 256 bytes. La página cero va desde $0000 hasta $00FF, la uno desde $0100 hasta $01FF, etc.

4.—Las zonas de memoria que están vacías no contienen nada (que pueda ser considerado como programa, puesto que se rellenan al azar). Al desensamblarlo, el ordenador interpreta algunos contenidos como mnemotécnicos de ensamblador, y se muestran en pantalla.

5.—Las tres interrogaciones "???" corresponden a cualquier valor (que se toma como instrucción) que no tiene mnemotécnico de ensamblador asignado.

DISCRIMINACION



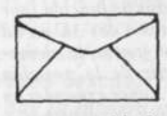
Quisiera deciros que poco a poco vais relegando al Vic-20 a un segundo plano, y eso se ve en los programas que estáis poniendo en la revista, puesto que los más interesantes quizá bajo mi punto de vista sólo son para el C-64, pongamos como ejemplo: "Contabilidad, Fútbol, Castellano para el C-64..." y los que tenemos un Vic-20 la mayoría de las veces nos tenemos que conformar con las colaboraciones que mandan los suscriptores (más o menos buenas) y creo que eso de cara a los usuarios del Vic-20 es una gran discriminación, puesto que yo adquirí hace poco un libro de programas para el C-64 que era "Aplicaciones para la casa y los pequeños negocios" y modificándolos van perfectamente en el Vic-20. También la "Base de Datos" que publicasteis en el número 3 del mes de abril está mejor la del C-64, por lo menos cara a la presentación en pantalla. Son algunos ejemplos que os pongo.

Otra cosa que me parece mal es que los programas que ponéis para el Vic-20 son casi todos para la memoria standard, cuando yo he recogido alguna opinión de usuarios del Vic-20 y hay mucha gente que ya tiene ampliación de memoria, ya sea de 8 ó de 16K, o quizá las dos. Además hay otro problema, cuesta mucho adaptar el programa a otras configuraciones de memoria, o no se puede, y si se puede hay que andar poniendo y quitando ampliaciones de memoria cada vez que usamos algún programa de estos.

Espero que me disculpéis por esta pequeña bronca.

César Guerrero López
C/Santucho, 73. 4-C
48006 Bilbao

COMPILADORES Y DESENSAMBLADORES



Quisiera que me aclararais las siguientes dudas que tengo:

1. ¿Qué es un compilador, un desensamblador y un assembler?

2. ¿Existe algún programa que convierta el Basic a lenguaje máquina?

3. Con el cartucho Monitor de Lenguaje Máquina (Vic-20) ¿Cómo se puede convertir un programa Basic a código máquina?

4. El programa "Constructor de datos" (número once) si se adapta al cassette ¿podría funcionar bien? ¿En qué se deberían hacer los cambios?

P.D. Felicitación a Errazquin por los dibujos.

Javier Recio Lamata
C/Fernán Caballero, 20
41001 Sevilla

1. Un compilador es un programa que convierte un programa Basic en otro idéntico pero en Lenguaje Máquina. Un desensamblador es un programa que sirve para poder "listar" un programa en código máquina, y un ensamblador lo contrario, sirve para escribir programas en c.m.

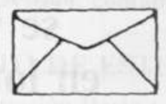
"Assembler" —traducido como ensamblador— es un lenguaje simbólico que con un programa ensamblador se traduce a código máquina.

2. Los compiladores (no conocemos ninguno para el Vic).

3. No se puede, el Monitor de Código Máquina del Vic es sólo ensamblador y desensamblador.

4. El programa original utiliza un tipo de ficheros que no se pueden utilizar en cinta. De todos modos, se podría intentar algo parecido con ficheros secuenciales, pero es bastante complicado. Otra posibilidad es hacer un programa que lea los datos, los imprima en la pantalla y utilizando el buffer del teclado, hacer que el ordenador se detenga, "pulse" la tecla return y vuelva a algún punto del programa. El inconveniente de este sistema es que se pierden variables (al introducir una línea nueva), pero se puede solucionar utilizando las posiciones de memoria como variables —haciendo POKE y después PEEK—, este segundo sistema puede resultar más sencillo que el anterior.

EXPANSOR PARA EL VIC-20



Me gustaría hacerles algunas preguntas sobre el expansor del Vic-20. Sólo poseo el cartucho de 16K ram y el Super expander con 3K ram. Cuando conecto el cartucho de 16K ram, el Vic asciende a 19967 bytes y conectando el Super expander a 6519. Y cuando conecto ambos cartuchos da 19381 bytes, funcionando todas las funciones del Super expander.

SEAMOS PREGUNTONES ¿?



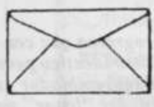
Mi pregunta es la siguiente: al tener conectados ambos cartuchos ¿la memoría debería ascender a 26486 bytes debido a los 3K ram del Super expander? o, ¿es normal este descenso a 19831 bytes, para poder usar las funciones y pantalla del Superexpander?

Cirilo Sánchez Barrado
C/Cedro, 4 - bajo-3
Cornella de Llobregat (Barcelona).

El cartucho de 16K provee al Vic unos 20K ram de memoria —Los 3583 de la inversión base más los 16K (16384 bytes) son los 19967 bytes que aparecen en la presentación —. El Super expander "toma" algunos bytes de memoria para definir las teclas de función (por eso la suma de los 3583 bytes + los 3K no da exacta). Cuando se conectan juntos, la memoria de 16 se coloca a partir de la posición $2000 (hasta la $4000), y el Super expander en la $A000 —donde la expansión de cartuchos—

Los 3K del Super expander quedan inutilizados para Basic, porque quedan en una posición no accesible (si se pueden utilizar cuando la ampliación es de menos de 8K), de modo que la memoria libre es la misma que si se tienen sólo los 16K. Como además el Super expander toma algunos bytes, el resultado es menos de los 19967 bytes.

PROGRAMAS CON JOYSTICK



Soy lector de su revista "Commodore World", aunque sólo tengo cuatro ejemplares (en los que no he tenido ninguna pega), me han enseñado muchas cosas, como sus Pokes misteriosos, etc.

La única pega que tengo es que, después de leer la explicación de algunos programas (casi siempre los mejores), suele aparecer la palabra JOYSTICK, y se me cambia la cara de color ya que no dispongo de uno. Me ha pasado varias veces lo mismo, en tres o cuatro programas, en los que por mucho que cambie no funcionan, en especial me refiero al juego "Alfa Ceti" de la revista número doce y les pido que me digan las instrucciones necesarias para que se pueda manejar desde el teclado.

Ignacio Calleja Pascual
C/San Antón, 65. 1º-Izq.
Pamplona (Navarra)

Las líneas a cambiar para que el programa funcione con el teclado son las siguientes:

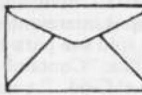
```
55 KY=PEEK (197): D=—1* (KY=17)
   —2* (KY=18) —4* (KY=52) — 8*
   (KY = 21)
60 F= (PEEK (653)<>1): P=P+1
70 IF D = 0 THEN 150
80 IF D = 1 THEN X1 = X1-1 : Y1 = Y
90 IF D = 2 THEN X1 = X1+1 : Y1 = Y
100 IF D = 4 THEN Y1 = Y1-1 : X1 = X
110 IF D = 8 THEN Y1 = Y1+1 : X1 = X
```

Las líneas 120 y 130 se suprimen. Las teclas de control son: A, D izquierda-derecha, O, L arriba-abajo y SHIFT para disparar.

Estas teclas pueden cambiarse variando los valores de la línea 55 (son los de la posición 197 al pulsar una tecla).

El inconveniente de usar este sistema es que no se puede mover en diagonal.

TAPAS PARA ENCUADERNAR



No sé si habrán recibido otras preguntas en el mismo sentido, pero mis sugerencias son:

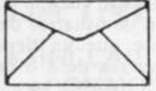
1. ¿No han pensado ustedes en editar unas tapas para encuadernar las revistas (por ejemplo en grupos de diez) y así poder coleccionarlas y guardarlas debidamente?

2. ¿Podrían incluir una sección editada en capítulos sobre un curso de informática?

Ramón Falip Espona
C/General Prim, 12
Belluis (Lérida).

Las tapas para encuadernar las revistas saldrán próximamente al mercado, tal y como habíamos anunciado, lo de los cursos lo vemos más difícil, porque ocuparía mucho espacio.

SUGERENCIAS



Os escribo esta carta porque creo interesante una idea que se me acaba de ocurrir, es la siguiente:

Parece que si bien los lectores no se animan mucho a colaborar en mandar colaboraciones, siempre son los mismos los que hacen sus apariciones y como dicen algunos lectores, esto puede ser causa del bajo nivel de conocimientos, entre los cuales me incluyo. Esto podría ser solucionado mediante unas explicaciones algo más detalladas en la revista y, aquí viene la idea, mediante un libro (del tipo del especial) en el cual se expliquen las cosas, se hagan cuadros más claros, se expliquen los pokes, incluso línea por línea en los primeros programas.

Lo fundamental es la teoría, bien explicada, más tarde se apoya en un programa el cual se podría entender mejor. Al final debería llevar un glosario.

El libro trataría de temas que no vienen claros en el manual, temas que salen en los programas de vuestra revista y que no hay espacios para explicaciones.

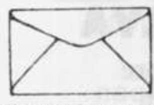
Esto se me ha ocurrido (no seré el único) ante la imposibilidad de hacer un pequeño juego, o de entender alguno de los programas de COMMODORE WORLD y de ver cómo gente de igual o menor edad que yo domina el Basic mucho mejor que yo.

Poseo un Vic, son unos cuatro años el tiempo que hace que lo tengo y si bien no es mucha la dedicación, puesto que estoy estudiando, y el manual está en inglés, he trabajado duramente para entender algo y algo si lo he entendido, pero QUIERO PROGRAMAR UN JUEGO SENCILLO, HACER PROGRAMAS TECNICOS y ayudar a la gente como yo (me gusta ayudar a los demás, este es mi defecto) no pido ser el mejor programador de España ni que mi juego se venda mucho (lo regalaría a la revista). Otra cosa, al Vic lo tenéis abandonado y todavía tiene que durar algo ¿no?

Animo con el comix, está muy bien, y a por el libro.

José María Martínez García
Apdo. 303
Cartagena.

MANUALES



Les rogaría, si ello fuera posible, me indicaran si Vds. pueden facilitarme, pagando su precio, el manual de la unidad de discos 1541, ya que carezco de él, al haberla adquirido usada. No me importaría que fueran fotocopias.

J. M. Alemany
C/Fray P. Vives, 37-13
46009 Valencia.

Los manuales de los ordenadores y de los periféricos los tiene Micro Electrónica y Control en Barcelona. Seguramente ellos podrán proporcionarte uno. El teléfono es el (93) 325-50-08.



Magia

La MAGIA son trucos, la MAGIA es divertida.

La MAGIA es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La MAGIA es una sección mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

MAGIA habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enviárnoslos comprobados, pues hay varios incorrectos. ¡Ah! y no nos mandéis trucos repetidos... ¡Listos!.

EL REY Y EL PASTOR

¿Conocéis la historia de un pastor que pidió un grano de arroz por la primera casilla de un tablero de ajedrez, el doble por la segunda, y así sucesivamente?

¿Conocéis ese número tan fantástico que no supieron encontrar entre todos los matemáticos de la época?

Pues bien, la solución que se buscó durante siglos os lo dará este pequeño programa.

```
1 A=2:FORN=1TO63:A=A*2:NEXTN:PRINTA
```

¿Increíble, no?

De haber sabido el rey a lo que se exponía, seguro que no hubiese aceptado.

¿No creéis?

Luis Angel Pifarre Gutiérrez
C/Bovila, 4, 3º 2.ª
Tel. 241 64 09
08004 Barcelona

JEROGLIFICOS

Hola amigos de Commodore:

Os mando este programa de una línea por si creéis conveniente publicarlo en vuestra sección de magia. Es un programa generador de jeroglíficos egipcios, si lográis descifrar alguno, tal vez encontréis un tesoro escondido.

¡SUERTE!

```
10 PRINT"(CLR)":FORA=1024TO2023:X=INT(RND(1)*150)+64:
POKE A,X:NEXTA.
```

Margarita Mercadal
Pompeu Fabra, 36
Palafrugell
(GIRONA)

DESACTIVA TECLAS

Os envío el siguiente truco que he descubierto: Poniendo PRINT CHR$ (8) se anulan las teclas shift y comm. Es un truco muy útil cuando se usan caracteres programables y sin querer (o queriendo) se da a dichas teclas.

PRINT CHR$(9) anula lo anteriormente dicho.

Javier Recio Lamata
C/Fernán Caballero, 20
41001 Sevilla

DOS TRUCOS UTILES

1. Consiste en poner los comentarios detrás de algún "goto" o "then nnnn" separado por dos puntos. De esta manera os ahorramos el byte de la sentencia REM.

Ejemplo: goto nnnn: comentario.

2. Si tenemos un programa en la memoria y queremos grabarlo en cinta detrás de otro, pero no sabemos donde acaba este, la solución consiste en rebobinar la cinta, escribir "verify" y en cuanto aparezca "verify error" ya podemos grabar nuestro programa.

Miguel del Pino González
C/Talavera, 7
28016 Madrid

CAMPANA DE GAUSS

Colegas vic-ciosos, os mando a continuación un programa de una sola línea que dibuja en la pantalla la llamada campana de Gauss:

10 Print "[CLR]": Poke 36879, 233: forx=0to21: Poke 7922+×22*int (11*cos(x*) 10.5)), 42: next

Si queréis que dibuje una línea horizontal cambiad el COS por COT.

También os mando unos POKES

POKE 788, 194 anula la tecla STOP

POKE 37159, 250 hace que el cursor tarde al responder las teclas del cursor y además hace que se mueva mucho más lento.

Julián María Manglano Garcia, 12 años
C/Acero de Canasteros, 16-4-A
18012 GRANADA

Nota: Aprovechando el último de tus trucos, vamos a darle otro uso que hemos descubierto aquí en la redacción.

Los que tengáis una impresora MPS-801 conectada a vuestro VIC habréis observado que a veces se queda "colgada" al sacar listados (sobre todo si éstos son largos). Esto se debe a que esta impresora está diseñada en principio para el C-64. Como el VIC es un 25% más RAPIDO que el C-64 en la mayoría de las cosas, envía datos tan rápidamente que la pobre impresora no es capaz de recibirlos al mismo ritmo. El truco consiste en hacer POKE 37159,2 (en vez de un dos se puede poner un número del uno al diez, el que veáis que mejor funcione). De este modo todas las funciones del VIC se retardan, perdiendo velocidad y "ajustándose" a la de la impresora.

FRACCIONES

Un problema que se presenta en muchos programas científicos es que los resultados numéricos vienen en forma decimal.

A continuación la solución: Conversión de números decimales a fracciones.

```
5 input "introduzca el n. decimal"; d
10 w=a=d int(d).
20 x=(1/a)-int (1/a): j=int (1/a)*w: w=j+O:o=j/int(1/a)
30 e=int (((int(d*w+.5)/w)-d)*1e9+a=x:if e then 20
40 print int(d*w+.5)"[crsr  ]/";:v=int (w+.5)
50 print int(d*1e9+.5)/1e9
```

David Cervigon Fernández
C/Badalona, 103
28034 Madrid

LETRAS DE POLVO

Os envío un truco de magia para que veáis lo que pasa. Yo lo he titulado letras de polvo. ¡¡Hasta la próxima!!

```
10 Print "[clr]"
20 Print "[7 crsr r] [5 crsr d] MANOLO"
30 goto 10
```

Luis Crespo (9 años)
Avda. Mediterráneo, 28 1-A
Ciudad Badía (Barcelona)

GRABAR EN CINTA

No es necesario avanzar una cinta que está rebobinada para grabar en ella. El ordenador graba unos segundos con un pitido agudo antes de salvar el programa.

T.C. RUN EE.UU.

PROGRAMOTECA

Si grabas en una misma cinta programas de varios tipos, seguramente tardarás mucho en encontrarlos. Es más util que uses varias cintas para diversos tipos de programas. Por ejemplo, una cinta para juegos y otra para programas de utilidad. Te divertirás más viendo como aumenta tu colección de un modo ordenado.

S.M. RUN. EE.UU.

PROBLEMAS CON LOS DISCOS

Cuando un disco floppy ha estado sometido a fríos muy intensos, puede llegar a causar problemas de carga o grabación. Si grabas en un disco frío, puede que lo que hayas grabado no funcione con un disco que esté caliente. Por lo tanto, guarda tus discos en un lugar que no esté ni muy frío ni muy caliente. Te ahorrarás problemas.

T.S. RUN EE.UU.

PROTEGIENDO LOS DISCOS

Para proteger los discos de polvo, suciedad, etc., es conveniente guardarlos en fundas de plástico individuales. Estas fundas no suelen costar mucho y son transparentes e impermebales.

E.M. RUN EE.UU.

POR LAS DOS CARAS

Es posible usar las dos caras de un disco floppy. Todo lo que hay que hacer es recortarle una nueva protección en la parte posterior.

Esta protección es lo único que impide que el disco pueda ser grabado por esta cara.

Hay que cortar la protección en el mismo lugar en el que estaría si fuera un disco normal. Para hacerlo puede uno ayudarse superponiendo otro disco con la cara hacia abajo.

D.W.V. RUN EE.UU.

DETECCION DE TECLAS

La posición de memoria 197 detecta cuál es la tecla que está pulsada en este momento. Esto puede usarse durante los programas. Por ejemplo: si el PEEK (197)=64, quiere decir que no hay ninguna tecla pulsada. Si es 32 que está pulsado el espacio, etc.

Para saber que número corresponde a cada tecla: 1 Print Peek (197): goto 1

T.H. RUN EE.UU.

SAVE & REPLACE

El comando de la unidad de discos 1541 SAVE "©0": NOMBRE" es muy util cuando estamos depurando un programa, pero a veces causa problemas, como borrar el programa, o incluso toda la información del disco. Para evitarlo podemos usar esta pequeña rutina:

```
59999 end
60000 open 15,8,15, "s0:nombre": clse 15: save "nombre" 8
```

Cada vez que quieras grabar la última versión de tu programa, teclea RUN 60000.

J.R. RUN. EE.UU.

DESABILITACION DEL TECLADO

En el VIC-20 o en el C-64, POKR 649,0 hace que el teclado deje de funcionar. Para volver a la normalidad: POKE 649,10

D.S. RUN EE.UU.

ALGUNOS POKES INTERESANTES

En el VIC-20:

POKE 808,114: Desactiva la tecla STOP/RESTORE todavia funciona.

POKE 818,100 o POKE 808, 127: Desactiva STOP, STOP/RESTORE y LIST.

POKE 808,112 devuelve a la normalidad.

POKE 818,73: POKE 819,245: Inhabilita el comando SAVE. Pulsando STOP/RESTORE se vuelve a la normalidad.

POKE 775,0: Inhabilita el comando LIST.

POKE 775,199: Lo habilita de nuevo.

L.F.S. RUN EE.UU.

MAS POKES INTERESANTES

En el C-64:

POKE 808,239: Desactiva STOP, pero la secuencia STOP/RESTORE todavía funciona.

POKE 808,225: Hace lo mismo, desactivando también STOP/RESTORE y LIST.

POKE 808,237: Vuelve a la normalidad.

POKE 775,200: Inhabilita el comando LIST.

POKE 775,167: Lo habilita de nuevo.

L.F.S. RUN EE.UU.



C-64

Generador de datas

José M. Beltrán Ortí
C/ Navarra, 3 - 1º 2ª
12580 Benicarlo (Castellón)

Después de varios días de trabajo ya tenemos nuestro programa en código máquina terminado y funcionando (aunque esto último no siempre), ahora sólo nos queda la FACIL tarea de pasarlo a Básic mediante unos cuantos DATAS.

La cuestión parece ser sencilla, de hecho lo es, pero empezamos a convertir cada uno de los bytes a decimal, a colocarlos en sentencias DATA y cuando aún no estamos por la mitad desconectamos el ordenador y lo mandamos 'a paseo'.

¿La razón? Es muy simple. Sencillamente el trabajo es arduo y pesado. Convertir los códigos en hexadecimal, escribir cientos de datas, números de línea, etc., no es un trabajo entretenido ni descansado.

Por qué no dejamos que sea el ordenador quien se encargue de ello.

El siguiente programa convierte a nuestro ordenador en un experto constructor de DATAS. Su estructura es la siguiente:

10-90: Estas líneas se encargan de preguntar y escribir el nombre del programa y las direcciones de inicio y final del programa en código máquina a convertir.

100-150: Estas se encargan de ir escribiendo los números de línea y los DATAS correspondientes con los valores de la posición de código máquina correspondientes.

160: Esta línea es la principal. En primer lugar, sitúa el cursor en 'HOME' y llena todo el 'Buffer' del teclado de 'RETURNs' con lo que las líneas que el programa ha creado pasan a formar parte de él.

170-200: Estas se encargan de borrar este programa y dejar solamente el nuevo programa con los Datas correspondientes.

Aunque me he olvidado de decirlo, este programa también crea la línea de lectura 'FOR N=...TO...:READ A:POKEN,A:NEXT' con lo que el trabajo a hacer es mínimo.

```
10 INPUT"[CLR]NOMBRE[SPC]DEL[SPC]PRO
GRAMA";A$:A$=A$+"[10SPC]":A$=LEFT$(A
$,10)
20 INPUT"DIRECCION[SPC]DE[SPC]INICIO
[SPC]DEL[SPC]C.M.";PO:INPUT"DIRECCIO
N[SPC]FINAL[SPC]DEL[SPC]C.M.";FI
30 CH=1000:PRINT"[CLR][CRSRD]":PRINT
CH"REM*******************************"
:CH=CH+10
40 PRINTCH"REM*[27SPC]*":CH=CH+10
50 PRINTCH"REM*[9SPC]"A$"[8SPC]*":CH
=CH+10
60 PRINTCH"REM*[27SPC]*":CH=CH+10
70 PRINTCH"REM**********************
*******":CH=CH+10
80 PRINTCH"FORN="PO"TO"FI":READA:POK
EN,A:NEXT":CH=CH+10
90 GOTO150
100 REM
110 PRINT"[CLR][CRSRD]":FORL=1TO7:PR
INTCH"DATA";:FORN=POTOPO+15
120 PRINTMID$(STR$(PEEK(N)),2)",";:I
FN=FITHENN=PO+15:L=8
130 NEXT:PRINT"[CRSRL][SPC]":CH=CH+1
0:PO=PO+16:NEXT
140 IFPO=>FITHENPRINT"RUN170":GOTO16
0
150 PRINT"100PO="PO":FI="FI":CH="CH:
PRINT"RUN100"
160 PRINT"[HOM]";:POKE198,9:FORN=631
TO640:POKEN,13:NEXT:END
170 N=0
180 PRINT"[CLR][CRSRD]":FORM=NTON+60
STEP10:PRINTM:NEXT
190 IFM<210THENPRINT"170N="M:PRINT"R
UN170"
200 GOTO160
```

VIC-20 SIN EXPANSION

La máquina de Galton

(Un experimento matemático)

Antonio Liz Femenias
C/ Federico Pareja, 58
Ciutadella (Menorca)

Hace más de 3 años que tengo el VIC-20, soy fiel seguidor de la revista desde el primer número y, sin embargo, es la primera vez que me decido a escribir viendo que se piden programas que no sean juegos. Pues bien, ahí va uno.

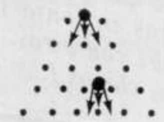
He procurado elegir un programa que puedan usar la mayoría de los lectores de la revista. Está escrito para funcionar directamente en el VIC-20 sin ampliación, pero puede hacerse funcionar en el C-64 con las siguientes modificaciones:



—Poner el cursor blanco (por ejemplo) en la línea 170.
—Modificar o quitar la presentación y explicación.
—Quitar las líneas 610, 740, 750 y 760.
—Sustituir las líneas 160 y 320 por estas:
160 CO=3 (color de las bolas y del histograma)
320 FI=40:P=1024:C=54272

El programa es lo que yo llamo un experimento matemático sobre la frecuencia de ciertos fenómenos (es decir, sobre el número de veces que se producirán).

La máquina presentada en este programa fue ideada por GALTON (su primo DARWIN se ha hecho más famoso). Consiste en un tablero vertical con clavos dispuestos formando un triángulo, como indica la figura, por cuyo vértice superior se dejan caer bolitas.



Las bolitas, al chocar con un clavo, pueden elegir entre dos caminos, (a la derecha o a la izquierda del clavo) igualmente probables, por eso, en cada bifurcación, lo más seguro es que la mitad de las bolitas vayan hacia la derecha y la otra mitad hacia la izquierda.

Veamos qué pasaría al lanzar 8 bolitas (8 = 2↑3 para que se pueda dividir por 2 tres veces seguidas) en la siguiente máquina:

En la primera bifurcación irán 4 hacia cada lado. Al bajar al siguiente nivel se volverán a dividir, irán 2 a cada pasillo, pero en el pasillo central coincidirán 2 bolitas procedentes de cada uno de los pasillos que tiene encima y, por tanto, recibirá 4 bolas.

Al llegar al último nivel se dividen de nuevo teniendo en cuenta que los pasillos centrales reciben bolitas procedentes de dos pasillos que tiene encima.

8
4 4
2 4 2
1 3 3 1

La distribución que se obtendrá PROBABLEMENTE en los cuatro pasillos finales será *1, 3, 3 y 1.*

En la máquina del programa con 9 pasillos finales se obtendría la siguiente distribución: 1, 8, 28, 56, 70, 56, 28, 8 y 1.

Estas distribuciones llamadas binomiales se pueden obtener construyendo el conocido triángulo de Tartaglia o de Pascal, en el cual cada línea comienza y termina con un uno y cualquier otro número se obtiene sumando los dos que haya sobre él.

1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1

TOTAL = 8 = 2↑3

TOTAL = 256 = 2↑8

Estos números y estas distribuciones se obtienen en otros muchos fenómenos matemáticos (Binomio de Newton, combinatoria, etc.).

La forma del histograma resultante es parecida a la llamada campana de Gauss (en la teoría de errores).

Espero que el programa os guste a pesar del "rollo" matemático que he puesto en la carta.



```
10 REM*MAQ. DE GALTON*********A. LIZ*
****************3-ENERO-'85***********
*CIUTADELLA*****
20 REM***************************PRESEN
TACION***************************
30 PRINT"[CLR]";:FORI=1TO22:PRINT"[RVSON]
[SPC][RVSOFF]";:NEXT:FORI=1TO21:PRIN
T"[RVSON][SPC][RVSOFF][20SPC][RVSON]
[SPC][RVSOFF]";
40 NEXT:FORI=1TO21:PRINT"[RVSON][SPC]
[RVSOFF]";:NEXT:PRINT"[RVSON][CRSRL]
[INST][SPC][RVSOFF]";
50 PRINT"[HOM]"SPC(67)"LA[SPC]MAQUIN
A[SPC]DE[SPC]GALTON"SPC(93)"UN[SPC]E
XPERIMENTO[10CRSRR][CRSRD]MATEMATICO
"
60 PRINTSPC(72)"[CRSRU][SHIFTU][SHIFT*]
[SHIFTI][CRSRD][3CRSRL][SHIFT-]C[SHIFT-]
[CRSRD][3CRSRL][SHIFTJ][SHIFT*][SHIFTK]
[CRSRU][SPC][RVSON]A.[SPC]LIZ[RVSOFF]
[CRSRU][6CRSRL][6COMM@]"
70 FORI=1TO3000:NEXT
80 REM***************************EXPLIC
ACION***************************
90 PRINT"[CLR][CRSRD]SE[SPC]ESPERA[SPC]
QUE[SPC]DE[SPC]256[2SPC](=2↑8)[SPC]B
OLAS[SPC]LANZADAS[SPC]VAYAN[SPC]A[SPC]
CADA[SPC]PASILLO:[CRSRD]"
100 PRINT"1[SPC]8[SPC]28[SPC]56[SPC]
```



```
70[SPC]56[SPC]28[SPC]8[SPC]1↑[SPC]↑[2SPC]
↑[2SPC]↑[2SPC]↑[2SPC]↑[2SPC]↑[SPC]↑[SPC]
↑[SHIFT-][SPC][SHIFT-][2SPC][SHIFT-]
[2SPC][SHIFT-][2SPC][SHIFT-][2SPC][SHIFT-]
[2SPC][SHIFT-][SPC][SHIFT-][SPC][SHIFT-]
";
110 PRINT"[RVSON]1[SPC]2[2SPC]3[2SPC]
4[2SPC]5[2SPC]6[2SPC]7[SPC]8[SPC]9[RVSOFF]
[CRSRD]AQUI[SPC]SE[SPC]LANZARAN[SPC]
128[2SPC]Y[SPC]SE[SPC]OBTENDRAN[SPC]
MAS[SPC]0"
120 PRINT"MENOS[SPC]UNOS[SPC]RESULTA
DOS[SPC]MITAD[SPC]DE[SPC]ESTOS[SPC]V
ALORES[CRSRD]HAY[SPC]DOS[SPC]VELOCID
ADES"
130 PRINT"QUE[SPC]SE[SPC]INTERCAMBIA
N[3SPC]PULSANDO[SPC]UNA[SPC]TECLA":P
RINT"[CRSRD][3SPC][RVSON]PULSA[SPC]U
NA[SPC]TECLA[RVSOFF]";
140 GOSUB720
150 REM*************************COMI
ENZO**************************
160 SO=36875:POKESO+3,15:POKE36879,2
9:CO=2
170 PRINT"[CLR][BLU][SPC]MAQUINA[SPC]
[COMMM][SPC][COMMG]":PRINT"[3SPC]DE"
180 PRINT"[SPC]GALTON[SPC][HOM][4CRSRD]
[SPC][RVSON]BOLA[RVSOFF][CRSRU][4CRSRL]
[4COMM@][HOM]";
190 FORI=1TO9:PRINTTAB(16)STR$(I)TAB
(18)"=":NEXT:PRINT"[HOM]"
200 PRINT"[HOM]";:FORI=1TO9:PRINTTAB
(16)"[COMMM]":NEXT:PRINTTAB(17)"[4COMMT]
[CRSRL][INST][COMMT][HOM]"
210 REM**************************DIB
UJO**************************
220 FORI=1TO8
230 PRINTTAB(10-I)"[SHIFTN]"SPC(2*I-
1)"[SHIFTM]"
240 PRINTTAB(9-I)"[COMMM][SPC]";:FOR
J=1TOI:PRINT"[SHIFTZ][SPC]";:NEXT:PR
INT"[COMMG]"
250 NEXT
260 B$="[SPC][COMMM][SPC][RVSON][COMM@]
[RVSOFF][SPC][RVSON][COMM@][RVSOFF][SPC]
[RVSON][COMM@][RVSOFF][SPC][RVSON][COMM@]
[RVSOFF][SPC][RVSON][COMM@][RVSOFF][SPC]
[RVSON][COMM@][RVSOFF][SPC][RVSON][COMM@]
[RVSOFF][SPC][RVSON][COMM@][RVSOFF][SPC]
[COMMG]"
270 FORI=1TO5:PRINTB$:NEXT:PRINT"[SPC]
[RVSON][SPC]1[SPC]2[SPC]3[SPC]4[SPC]
5[SPC]6[SPC]7[SPC]8[SPC]9[SPC][RVSOFF]
[HOM]"
280 X$="[22CRSRR]":Y$="[HOM][17CRSRD]
":G=99
290 REM*************************CAID
A BOLAS**********************
300 FORH=1TO128
310 PRINT"[HOM][6CRSRD][CRSRR]";H
320 FI=22:P=7680:C=30720:A=P+10:Z=P+
16*FI
330 FORJ=1TO8
340 GOSUB650
350 A=A+FI
360 GOSUB650
370 IFRND(1)>.5THENY=FI-1:GOTO390
380 Y=FI+1
390 A=A+Y
400 NEXT
410 F=((A-Z)/2)
420 C(F)=C(F)+1
430 PRINTLEFT$(Y$,F)LEFT$(X$,19);C(F
)
440 DD=42
450 IFC(F)<DDTHENGOSUB650
460 DD=DD-8:IFDD<0THEN500
470 A=A+FI
480 GOTO450
490 REM*******************DIBUJO HI
STOGRAMA*******************
500 N=INT(C(F)/8)
510 R=C(F)-N*8
520 NN=1
530 IFN>=NNTHENPOKEA,160:POKEA+C,CO:
A=A-22
540 NN=NN+1
550 IFNN<6THEN530
560 IFN=6THEN630
570 FORD=1TOR:READE:NEXT:RESTORE
580 DATA100,111,121,98,248,247,227
590 IFR=0THEN610
600 POKEA,E:POKEA+C,CO
610 GOSUB750
620 NEXT
630 GOSUB720
640 RUN
650 REM*****************DIBUJA Y BOR
RA BOLA*********************
660 POKEA,81:POKEA+C,CO
670 GETA$:IFA$<>""THENG=G-69:IFG<0TH
ENG=99
680 FORI=1TOG:NEXT
690 POKEA,32:POKEA+C,CO
```



```
700 RETURN
710 REM**************************ESPER
A TECLA*************************
720 A$="":GETA$:IFA$=""THEN720
730 RETURN
740 REM******************************SO
NIDO*************************
750 POKESO,200:FORI=1TO30:NEXT:POKES
O,0
760 RETURN
```

NOTA: El código [INST] que aparece en alguna de las líneas se obtiene pulsando simultáneamente Shift y la tecla Inst/Del.

C64

Rutina de trazado

José M. Beltrán Ortí
C/ Navarra, 3 - 1º 2ª
12580 Benicarló (Castellón)

El momento de depurar un programa puede convertirse en una tortura muchas veces y mandar 'al traste' toda una sesión de trabajo.

Por suerte nuestro ordenador se encarga de avisarnos cuando hemos cometido un error, parando el funcionamiento e indicándonos en qué línea se ha cometido.

Pero muchas veces esto no es suficiente porque el problema podemos tenerlo muchas líneas antes, o lo que es peor, el ordenador no detecta ningún fallo y, sin embargo, el programa no se comporta como nosotros queremos.

El siguiente programa nos permite visualizar el número de línea que se está ejecutando en cada momento y ayudarnos a descubrir dónde está el fallo que queremos detectar.

Esta rutina opera en código máquina, por lo que una vez ejecutado el programa no ocupa espacio en Basic y, por tanto, es compatible con cualquier otro programa Básic.

El programa en Basic creo que no es necesario comentar dada su simplicidad. En cuanto al programa en código máquina su funcionamiento es el siguiente:

$C000-C018: Se encargan de copiar el Basic de la ROM en la RAM que se encuentra debajo de esta.

$C019-C027: Cambia dentro de la rutina [RUN] del Basic para que pase a esta rutina de trazado a la dirección $C02E.

$C028-C02D: Desconecta la ROM del Basic, a partir de este momento el Básic se encuentra en RAM.

$C02E-C033: Comprueba si la entrada es directa o bien está dentro de un programa. Si es directa salta al final de la rutina no ejecutándola.

$C034-C03B: Guarda la actual posición del cursor.

$C03C-C052: Coloca el cursor en la parte superior derecha y borra el espacio que ocupara el número de línea por el programa de trazado.

$C053-C057: Escribe el símbolo '#' que precederá al número de línea.

$C058-C05E: Convierte el número de línea (en forma de Byt Bajo/Byt Alto) a un número decimal entero y lo escribe.

$C05F-C06F: Mira si se ha pulsado la tecla 'COMMODORE' y si es así genera un retardo que permite seguir mejor el trazado.

$C070-C077: Coloca el cursor en la posición que estaba inicialmente.

$C078-C07A: Continúa con la rutina [RUN] que se había interrumpido.

```
10 REM****************************
20 REM*                          *
30 REM*         T R O N          *
40 REM*                          *
50 REM* J. M. BELTRAN - ENERO'85 *
60 REM*                          *
70 REM****************************
80 REM
90 REM
100 PRINT"[CLR][15CRSRR][RVSON]T[SPC]
R[SPC]O[SPC]N[RVSOFF]"
110 PRINT"[CRSRD]EL[SPC]PROGRAMA[SPC]
[RVSON]TRON[RVSOFF][SPC]INCORPORA[SPC]
```



```
UNA[SPC]NUEVA[4SPC]SENTENCIA[SPC]BAS
IC";
120 PRINT"[SPC]MEDIANTE[SPC]LA[SPC]C
UAL[SPC]APARECEEN[SPC]EL[SPC]BORDE[SPC]
SUPERIOR[SPC]DERECHO[SPC]DE[SPC]";
130 PRINT"LA[6SPC]PANTALLA[SPC]EL[SPC]
NUMERO[SPC]DE[SPC]LINEA[SPC]QUE[SPC]
SE[SPC]ESTA[SPC]EJECUTANDO[2SPC]";
140 PRINT"DE[SPC]UN[SPC]PROGRAMA."
150 PRINT"[CRSRD]ESTE[SPC]NUMERO[SPC]
DE[SPC]LINEA[SPC]APARECE[2SPC]PRECED
IDO[SPC]POR[SPC]EL[SPC]SIGNO[SPC]'#' "
160 PRINT"[CRSRD]LA[SPC]SENTENCIA[SPC]
[RVSON]TRON[RVSOFF][SPC]SE[SPC]ACTIV
A[SPC]MEDIANTE[4SPC][RVSON]SYS49152[RVSOFF]
";
170 PRINT"[2SPC]SE[SPC]DESACTIVA[SPC]
PULSANDO[SPC][RVSON]RUN/STOP-RESTORE
[RVSOFF]"
180 PRINT"[CRSRD]PARA[SPC]HACER[SPC]
MAS[SPC]LENTO[SPC]EL[SPC]PASO[SPC]DE
[SPC]LINEAS[2SPC]PULSAR[SPC]LA[SPC]T
ECLA[SPC][RVSON]COMMODORE[RVSOFF]"
190 PRINT"[CRSRD]LA[SPC]RUTINA[SPC]E
N[SPC]CODIGO[SPC]MAQUINA[SPC]QUE[SPC]
SE[SPC]OCUPADE[SPC][RVSON]TRON[RVSOFF] ";
200 PRINT"SE[SPC]ENCUENTRA[SPC]ENTRE
[SPC]LAS[SPC]POSICIONESDE[SPC]MEMORI
A[SPC]49152[SPC]Y[SPC]49274."
210 PRINT"[CRSRD]CUANDO[SPC]ESTES[SPC]
LISTO[SPC]PULSA[SPC][RVSON]RETURN[RVSOFF] "
220 GETA$:IFA$<>CHR$(13)THEN220
230 REM
240 REM
250 REM
260 REM
270 REM
1000 REM******************************
*
1010 REM*
*
1020 REM*       T R O N
*
1030 REM*
*
1040 REM******************************
*
```

```
1050 FORN= 49152 TO 49274 :READA:POK
EN,A:NEXT
1060 DATA169,0,133,251,169,160,133,2
52,160,0,177,251,145,251,200
1070 DATA208,249,230,252,165,252,201
,192,208,239,169,76,141,225,167
1080 DATA169,46,141,226,167,169,192,
141,227,167,165,1,41,254,133
1090 DATA1,165,123,201,8,48,68,56,32
,240,255,134,251,132,252
1100 DATA162,0,160,32,24,32,240,255,
162,8,169,32,32,210,255
1110 DATA202,208,250,200,24,32,240,2
55,169,35,32,210,255,166,57
1120 DATA165,58,32,205,189,173,141,2
,201,2,208,10,162,0,160
1130 DATA0,232,208,253,200,208,250,1
66,251,164,252,24,32,240,255
1140 DATA108,8,3
2000 NEW
```

```
., C000  A9 00     LDA #$00
., C002  85 FB     STA $FB
., C004  A9 A0     LDA #$A0
., C006  85 FC     STA $FC
., C008  A0 00     LDY #$00
., C00A  B1 FB     LDA ($FB),Y
., C00C  91 FB     STA ($FB),Y
., C00E  C8        INY
., C00F  D0 F9     BNE $C00A
., C011  E6 FC     INC $FC
., C013  A5 FC     LDA $FC
., C015  C9 C0     CMP #$C0
., C017  D0 EF     BNE $C008
., C019  A9 4C     LDA #$4C
., C01B  8D E1 A7  STA $A7E1
., C01E  A9 2E     LDA #$2E
., C020  8D E2 A7  STA $A7E2
., C023  A9 C0     LDA #$C0
., C025  8D E3 A7  STA $A7E3
```



```
., C028  A5 01     LDA $01
., C02A  29 FE     AND #$FE
., C02C  85 01     STA $01
., C02E  A5 7B     LDA $7B
., C030  C9 08     CMP #$08
., C032  30 44     BMI $C078
., C034  38        SEC
., C035  20 F0 FF  JSR $FFF0
., C038  86 FB     STX $FB
., C03A  84 FC     STY $FC
., C03C  A2 00     LDX #$00
., C03E  A0 20     LDY #$20
., C040  18        CLC
., C041  20 F0 FF  JSR $FFF0
., C044  A2 08     LDX #$08
., C046  A9 20     LDA #$20
., C048  20 D2 FF  JSR $FFD2
., C04B  CA        DEX
., C04C  D0 FA     BNE $C048
```

```
., C04E  C8        INY
., C04F  18        CLC
., C050  20 F0 FF  JSR $FFF0
., C053  A9 23     LDA #$23
., C055  20 D2 FF  JSR $FFD2
., C058  A6 39     LDX $39
., C05A  A5 3A     LDA $3A
., C05C  20 CD BD  JSR $BDCD
., C05F  AD 8D 02  LDA $028D
., C062  C9 02     CMP #$02
., C064  D0 0A     BNE $C070
., C066  A2 00     LDX #$00
., C068  A0 00     LDY #$00
., C06A  E8        INX
., C06B  D0 FD     BNE $C06A
., C06D  C8        INY
., C06E  D0 FA     BNE $C06A
., C070  A6 FB     LDX $FB
., C072  A4 FC     LDY $FC
., C074  18        CLC
., C075  20 F0 FF  JSR $FFF0
., C078  6C 08 03  JMP ($0308)
.
```

GLOSARIO

Data conversion - Conversión de datos. El proceso de modificación de los datos de un sistema de representación a otro.

Data density - Densidad de datos. El número de bytes por pulgada (bpi) almacenados en cinta magnética.

Data error - Error en discos. La desviación del valor teórico esperado, normalmente se produce en los cálculos aritméticos y suele ser causado por los redondeos en las conversiones a/de coma flotante.

Data File - Archivo de datos. Un grupo de datos guardados con un orden y relación determinada, por ejemplo los de una contabilidad.

Data flowchart - Diagrama de flujo de datos. Es un organigrama o diagrama de flujo que representa las operaciones a que son sometidos los datos en un programa para resolver un problema determinado.

Data integrity - Integridad o seguridad de datos. La seguridad de los datos frente a su destrucción accidental o intencionada, así como para evitar el acceso a personas no autorizadas.

Data library - Biblioteca de datos. Un conjunto de archivos de datos relacionados entre sí. Por ejemplo los datos de un control de stock.

Data management - Gestión de datos. La función de controlar la entrada, análisis, distribución, almacenamiento y recuperación de datos.

Data processing (DP) - Proceso de datos. La realización sistemática de operaciones en el tratamiento de los datos.

Debug - Depurar. La detección, corrección y eliminación de errores en un programa de ordenador.

Decimal alignment - Alineamiento decimal. El encolumnado de las comas decimales en los números representados en pantalla o impresora.

Default - Por falta. El valor alternativo que toma una variable en caso de falta u omisión.

Degradation factor - Factor de degeneración. La medida del deterioro de los datos en un medio magnético en función del tiempo y el uso.

Delay - Retardo. El tiempo que se retarda o detiene un proceso.

Delete (DEL) - Borrar. Quitar o retirar un carácter, dato o fichero de la memoria, programa o unidades auxiliares de memoria.

Delimiter - Delimitador. El carácter que indica la separación de los datos en una cadena (string). Puede ser un signo de puntuación u otro que no utilicemos normalmente.

Descriptor. Utilizado para indicar o catalogar el tipo de información que contiene un archivo, generalmente se coloca en el primer registro de éste.

Desktop computer - Ordenador de sobremesa. *Es otra palabra utilizada para referirse a un microcomputador.*

Development system. Sistema de desarrollo. Un sistema de microcomputador que sirve para facilitar el diseño y realización de programas y circuitos (software y hardware).

Development time. Tiempo de desarrollo. El tiempo que se tarda en preparar y depurar las rutinas o el hardware, es sinónimo de tiempo de programación.

Development tools. Herramientas de desarrollo. Son aquellos programas de utilidades que facilitan el diseño de programas y/o circuitos.

MARKETCLUB

Servicio gratuito para nuestros lectores *PARTICULARES*. Los anuncios serán publicados durante 1 mes. Debido a ciertos abusos que nos han sido comunicados, los anuncios gratuitos de Market Club *SOLAMENTE* serán publicados si vienen con nombre, apellidos y dirección completa.

MERCADILLO

• Se vende un Commodore VIC-20 de segunda mano prácticamente nuevo. El precio es de 30.000 pesetas pero en el precio está incluido el Datassette "1530", un padle, más de 15 revistas de ordenadores de todo tipo, y además de esto una suscripción en una revista de ordenadores y que trae todo tipo de novedades para el VIC-20 y otros. Dirigirse a: c/ Moncayo Perla, 1. Puerta Norte 5º E. Fuengirola (Málaga). Juan Manuel. ***(Ref. M-320).***

• Vendo Spectrum 48K con garantía un año desde 1 de febrero/85 con 8 juegos e instrucciones en castellano. ¡A estrenar! Sólo 35.000. Telf. 445 50 06 oficina. 445 95 30 particular. Miguel Angel. ***(Ref. M-321).***

• Vendo VIC-20 (comprado en el 84); cartucho de código máquina, cursos de introducción al Basic (1.ª y 2.ª parte); guía de referencia del programador; innovative computing VIC; algunos programas de utilidades y juegos; monitor de fósforo verde; todo junto o separado. Precio a convenir. Interesados escribir a: Rafael Balongo García, c/ Armengual de la Mota nº 1-9º A. 29007-Málaga. O llamar al teléfono 39 50 77 preferentemente por la mañana. ***(Ref. M-322).***

• Vendo VIC-20 por 25.000 ptas. (1-84). Regalo 2 libros de introducción lenguaje de programación Basic parte I y II con sus correspondientes cintas explicativas + cinta de programa de control de stocks + cinta con juegos. Vendo ampliación de memoria universal de 64K. (32 libres y 32 ocultas) por 15.000 ptas. (18.900,—). Llamar a cualquier hora a (973) 74 01 54. Jorge Molins. c/ Mayor, 24. Almacellas (Lérida). ***(Ref. M-323).***

• Vendo cartucho Simon's Basic con manual. Precio 13.000. Alejandro Vegter. c/ Virgen de Guadalupe, 20-6 F. 10001-Cáceres. Telf. (927) 24 59 74. ***(Ref. M-324).***

• Vendo VIC-20 más ampliación de 16K (comprado en julio del 84) por 30.000 ptas. Regalo libro "Acceso rápido al Vic-20", muchas revistas y tres programas de 16K en cinta. Todo como nuevo. Llamar de 3 a 4 al 447 02 71 de Madrid. Juan Carlos Herrero Antolín. c/ Bravo Murillo, 41-17, 4º 2. 28015-Madrid. ***(Ref. M-325).***

• Vendo Commodore 64. Comprado en julio de 1984, con funda, unidad de cintas Datassette, joystick, cuatro cintas de juegos, guía de referencia de programador y catorce revistas Commodore. Precio a convenir. Carlos Alvarez Villa. c/ Gomeznarro, 120. 28043-Madrid. Telf. 200 13 30. En horas de comidas o a partir de las 10 de la noche. ***(Ref. M-326).***

• Soy profesor de E.G.B. (psicólogo nacional) que trabaja en el campo de la Pedagogía Terapéutica y deseoso de intercambiar experiencias y conocimientos acerca de un mejor aprovechamiento de la informática en el proceso educacional. Poseo el Commodore 64, unidad de disco y Datassette, lenguajes: Pilot, Logo, Simons, Basic, Forth, Pascal, etc. y una buena biblioteca de programas que seleccionados estratégicamente pueden contribuir a mejorar las capacidades psicomotrices que inciden en la lecto-escritura y cálculo. Los interesados en estas actividades pueden escribir a: José García Madera. c/ Regla Sanz, 11. 41010-Sevilla. ***(Ref. M-327).***

• Compro Commodore y sus periféricos, programas, libros. Miguel Angel García García. c/ Lotería, 3. 48005-Bilbao. Telf. (94) 415 85 67. ***(Ref. M-328).***

• Vendo ordenador Júpiter ACE. Forth, el camino para aprender a programar en el lenguaje más veloz es poseer el ordenador más barato: el Júpiter ACE. Llamar por la noche al 254 80 62. Rafael Rojo Rojo. Avda. Pablo Iglesias, 50. Madrid-20. ***(Ref. M-329).***

• Desearía contactar con alguien que tuviera los cursos de introducción al Basic (I y II). Llamar al teléfono 476 52 54. Miguel Angel Ordóñez Alfonsín. c/ Antonio López, 149 1º D. Madrid. ***(Ref. M-330).***

• Vendo impresora Plotter 1520 marca Commodore, prácticamente sin usar; compatible con el Commodore 64, precio: 43.000; valor real 59.900. Ramón. Telf. (94) 469 45 92 de Vizcaya. c/ Tilos, 2-2º Dcha. Neuri (Vizcaya). ***(Ref. M-331).***

• Cambiaría controlador doméstico nuevo (un mes de uso), más un cartucho de simulación de conducción por cartucho de ampliaciones a 16K RAM para VIC-20, también pagaría hasta 6.000 ptas. por ella. También pagaría 4.000 ptas. por módulo de 8K y 3.000 ptas. por un super-expander; cada uno de ellos también los cambio por E.C. Doméstico. Escribir a: José Tomás Gómez Valencia. c/ María Díaz de Haro, 16-3º F. 48014-Bilbao. ***(Ref. M-332).***

• Vendo VIC-20 más cassette C2N por 28.000 ptas. Buen estado. Regalo juego y curso de Basic. Llamar hora de comer. Preguntar por José. Telf. 610 63 29. ***(Ref. M-333).***

• Vendo C-64 con 8 meses de uso, más CN2, juegos, joystick de Spectravideo, y un libro con 60 programas para el C-64 en inglés. Todo por 69.000 ptas. Antonio Torralba Pinedo. c/ Avda. Cayetana del Toro, 46-11K. Cádiz. ***(Ref. M-334).***

• Vendo VIC-20 (junio 84) por 27.000 ptas. Incluyo introducciones Basic I y II (4 cassettes), cartucho, cintas juegos (Ski-run, y Contabilidad personal), guía ayuda al programador y diversas cintas con juegos y aplicaciones. Llamar al 346 70 17 (9-10 noche) o escribir a Luis Sotorzano Vázquez. c/ Bartrina, 1, 3 Esc. B 3º 1. 08030-Barcelona. ***(Ref. M-335).***

• Vendo cartucho de ajedrez (sargon II Cheis) para VIC-20, nuevo por 2.500. Escribid a Juan M. Robles. c/Doctror Trueta, 31-32, 2º. Castedellfels. ***(Ref. M-336).***

• Vendo VIC-20, curso introducción al Basic, con varios programas por sólo 20.000 ptas. Llamar al (981) 23 56 12. Miguel Angel Ferantes Pérez. ***(Ref. M-337).***

• Me falta Commodore World nº 1, 3, 4, 5, 6 y tengo 10 juegos comerciales. Cambio una revista por juego (revista 315 ptas. y juego 2.500) y por el nº 1 tres juegos; imprescindible que las revistas estén en perfecto estado. Mi dirección es Juan Manuel Jurado. Plaza Villa, 10. Torredembarra (Tarragona). Escribidme, contesto a todos. ***(Ref. M-338).***

• Vendo ampliación de memoria 16K RAM para VIC-20, precio a convenir. Telf. 332 56 90 prefijo 93 de Barcelona. José. ***(Ref. M-339).***

• Vendo VIC-20 prácticamente nuevo (noviembre del 84) procedente sorte de TV3 por sólo 20.000 ptas. Regalo guía usuario de VIC-20. Heribert Farreny Agras. Elisa, 24-26. 1º 3 08023-Barcelona. Telf. 248 05 51 (93). Laborables por la tarde. ***(Ref. M-340).***

• Vendo VIC-20 con datassette, también ofrezco el cartucho de ajedrez sargon. Toda la colección de revistas de Commodore World, así como manual en castellano. El libro "Introducción al lenguaje de programación Basic", más dos cintas cassette curso. Lo ofrezco todo por unas 39.000 ptas. Con la compra de todo regalaría varios juegos en cintas. Joan Bonjoch Barguesoc. c/ Amílcar, 186. 08032-Barcelona. Telf. 347 95 07. ***(Ref. M-341).***

• Se vende VIC-20 con 40 columnas, 25 caracteres, 2 cartuchos utilidades, manejo cassette 6 veces más rápido, código máquina, editor de texto, calculadora hexadecimal/decimal, super-expander, base de datos, muchos juegos, catálogos información periféricos, guía del usuario, etc. Todo casi nuevo y barato por sólo 40.000 ptas. (a convenir). Máximo Martín. c/ Orillamar, 75-4º Izda. Telf. (981) 20 40 60. 15002-La Coruña. ***(Ref. M-342).***

• Vendo VIC-20. Datassette C2N. Monitor "Fontec" fósforo verde con sonido. Manual introducción al Basic con las cintas (1.ª parte). Cartucho de ajedrez. Revista Commodore. Todo en perfecto estado. Precio 45.000 ptas. Pedro Flores. c/ Santander, 7. Barbera del Vallés (Barcelona). Telf. 718 53 47. ***(Ref. M-343).***

• Vendo ordenador VIC-20 con manuales, curso de Basic con cintas, programas de RTTY y morse, esquemas para construir interfaces, interface construido para recibir y transmitir en morse y recibir RTTY, interface para conectar cualquier cassette al VIC-20. Libro 50 programas listados y extras, todo por 25.000 ptas. Jesús Prieto Freire. c/ Barrafón, 25. 28011-Madrid. Telf. 463 72 30. ***(Ref. M-344).***

• Deseo comprar una unidad de disco VIC 1541 a precio asequible, mi teléfono es el (977) 31 82 75. Juan Caballero Díaz. Barrio Gaudí, bloque amarillo, 3º nº 8. Reus (Tarragona). ***(Ref. M-345).***

• Cambio controlador doméstico de 8 relés, nuevo para VIC-20 o CBM-64 con manuales por unidad de cassette Commodore. También cambio cartucho para VIC-20 "Road Race" por joystick Commodore. Tomás Gómez Valencia. c/ María Díaz de Haro, 16 3F. 48013-Bilbao. ***(Ref. M-346).***

• Vendo Commodore 64, con cassette, joystick y juegos. Además del manual de la guía de referencia C-64. Todo por 55.000 ptas. También muchas revistas. Sergio Vernis Perraman. c/ Mavern, 11-5º 1. Barcelona. Tef. 209 39 64. ***(Ref. M-347).***

• Vendo consola de vídeo-juegos Phillips (aparato con dos joysticks incorporados) por 19.000 ptas. Regalo cuatro cartuchos: Pack-man (comecocos), Space monster (marcianitos), Take the money and run (Coja el disco y corra), y Baseboll (Baisboll). Todo está en perfecto estado. Interesados dirigirse a: Francisco Jorge Guillem. c/ Trinquet, 56-1º. Catarroja (Valencia). ***(Ref. M-348).***

• Se vende VIC-20 en perfecto estado (casi nuevo), con Dattassette, 5 cintas de juegos, "Manual del Usuario", "Introducción al lenguaje Basic: Parte I", libro para aprender BASIC, precio a convenir. Daniel Garriga, C.N. II, Walden-7, 2-51, Sant Just Desvern, Barcelona Tel. (93) 372 02 43 ***(Ref. M-349).***

• Me interesaría, que algún lector de esta revista, que posea los números 1 y 4, se ponga en contacto conmigo para sacar fotocopias de las mismas. Luis Fernando Otal. C/ Tapiola, 29. Barcelona. Tel. 242 02 98. ***(Ref. M-350).***

• Vendo programa en CM; monitor-desensamblador-ensamblador 2 pasos para VIC-20 + tarjeta 40180 (Terrenoret, Deltabit o compatible) + 8K RAM como mínimo. El monitor posee más de 20 comandos y el programa viene acompañado por su manual de funcionamiento. Precio: 4.000 ptas. Marcos Gregori. Gran vía Germanías, 10-Pta. 7. 46006 Valencia. ***(Ref. M-351).***

• Vendo 3 revistas extranjeras (RUN) de los EE.UU. por 600 ptas. cada una. Las 3 por 1.500. Tienen mucha información sobre clubs y software. Interesados escribir a: Javier Recio Lamata. C/ Fernán Caballero, 20. 41001 Sevilla. ***(Ref. M-352).***

• Vendo VIC-20 más accesorios: cassette CN2, superexpander más 3K, Sargon II Chess, Omerga Race, 1 joystick y 2 padles. Bastantes cintas (todas originales) de Indescomp, y Matrix de Aridnumer. Libros: First book of VIC, Second book of VIC games (24 juegos) de Computer. EE.UU. El curso de Basic con los programas en cinta y guía del usuario, esta en castellano. Todo por 65.000 ptas. José Antonio Márquez García. C/ Oscar Garballo, 5. 41005 Sevilla. Tel. 58 01 78. ***(Ref. M-353).***

• Vendo VIC-20 por 20.000 ptas. Regalo además cartucho Sargon II Chess, cartucho 3K RAM más superexpander 6.000, cartucho ayuda al programador 5.000, módulo de expansión de memoria para 6 cartuchos por 10.000 ptas. Jordi Joaquín Esteve. Passeig de la Independencia, s/n, bloque 1, escalera B, 3º A. 43005 Tarragona. Tel. 22 90 42. Llamar de 9 a 11 mañanas. ***(Ref. M-354).***

• Vendo VIC-20 con muchos programas excelentes, en perfecto funcionamiento por sólo 20.000 ptas. Llamar a Juan. Tel. 734 67 37. Cerro del Castañar, 167, 4. 28034 Madrid. ***(Ref. M-355).***

• Compro esquema completo de interface par CW-RTTY para VIC-20, con diseño del circuito impreso (fotocopias) primeramente escribir indicando qué integrados lleva, para ver si me interesa. También quisiera el esquema del circuito electrónico del VIC-20 completo. Intercambio programas del VIC. Julio Ogando Muradás (EA1BTD) Vía

Norte, 34 bajo (A.PARIS) Vigo-4 (Pontevedra). *(Ref. M-356).*

• Vendo VIC-20 en perfecto estado con muchos programas buenos y libros por sólo 20.000 ptas. Interesados dirigirse a Horacio. Tel. 245 69 09. Avda. Bon, 13 - 6º. Madrid. *(Ref. M-357).*

• Se está formando un club para el Commodore 64. Interesados llamar al teléfono (983) 33 26 52 de 2 a 3 de la tarde. Valladolid. Serafín Cuenca. C/ Jesús Ribero Meneses, 2 - 3º A. 47014 Valladolid. *(Ref. M-358).*

• Vendo Commodore VIC-20 (enero 84) por 25.000 ptas. 15 juegos en cintas a 12.000 ptas (precio actual de los 15 juegos 30.000 ptas.), cartucho "Ayuda al Programador" por 4.000 ptas, un cartucho de juego por 2.000 ptas. Ampliación de memoria a 16K por 10.000 ptas. VIC Speed por 5.000 ptas. Total: 58.000 ptas., si quieren comprar todo el lote, precio 35.000 ptas. (precio al que yo lo he comprado 91.000 ptas.). Preguntar por Sunaj. Tel. 254 04 61 (horas de comida) Casanova, 49 - 6º 2º Dcha. 08011 Barcelona. *(Ref. M-359).*

• Desearía comprar aumento de memoria de 16K y 64K para VIC-20. Módulo de expansión e impresora para VIC-20. Carles Bonjoch. Molen Amic, 1. Tel. 24 21 08. Horario de oficina. *(Ref. M-360).*

• Vendo cartucho de 16K por 9.500 ptas. negociables. José García Alegre. C/ De Eiximenis, 11. Parets del Vallés (Barcelona). Tel. 562 11 94 *(Ref. M-361).*

• Vendo VIC-20 (un año) como nuevo, libro "VIC-20 guía del usuario" y regalo un montón de programas por 25.000 ptas. (preferentemente zona Palma de Mallorca, ciudad). Juan José Peña Gómez. Garita, 16-3º. 2. 07015 Palma de Mallorca *(Ref. M-362).*

• Por cambio de ordenador vendo para VIC-20: cartucho ampliación de memoria de 16K (10.000 ptas); vídeo-juego en cartucho ALIEN (3.000 ptas); cartucho de ajedrez Sargon II Chess (3.000 ptas) y estoy interesado en programas en cinta o en disco para el Commodore 64. Manuel Fuentes Sorrivas. Salamanca, 11 - 4º C - Vigo-11 (Pontevedra). Tel. (986) 47 17 68. *(Ref. M-363).*

• Vendo VIC-20 (marzo-84) con ampliación 28K, superexpander, cintas de juego, cartucho ROAD RACE, todo por 40.000 ptas. y también regalo libros del VIC-20, Acce-Rapido, VIC-20 y curso Basic. Interesados a Juan Griñó. Casas y Amigo, 63 - 1º. Tel. 398 27 76. Badalona (Barcelona). *(Ref. M-364).*

• Vendo Music-64 (teclado con interface, diskette, cassette e instrucciones). Totalmente nuevo. Comprado el día 25 de enero del 85. Todo por 34.000 ptas. Llamar al tel. (976) 88 21 12. Calatayud. Preguntar por Pablo. *(Ref. M-365).*

• Urge vender VIC-20 con ampliación de 16K, con algunos juegos y el cartucho de ayuda al programador. Todo 22.000. Llamar (93) 674 62 19. Pepe de 21 a 23 horas. *(Ref. M-366).*

• Busco esquemas VIC-20, impresora, U. Discos, pago fotocopias, intercambio programas radioaficionados (CW-RTTY-ATMOR-QSL,s etc.). Dispongo gran surtido 3,5, 8, 16K: interesados dirigirse a: Antonio —EA 9 MY— García Cabrelles, 54. Melilla. *(Ref. M-367).*

• Vendo un VIC-20, más el libro "Acceso rápido al VIC-20", más el Manual del Usuario para VIC-20 con los dos libros de "Introducción al Basic para VIC-20", con los cuatro cassettes de programación, más un juego de ajedrez en cartucho y dos juegos en cassette. Todo por valor de 49.000 ptas. lo vendo por 33.000 ptas. Llamar al tel. (93) 421 20 08. Barcelona. Preguntar por Santiago Pla Bori de 14,30 a 15,30. *(Ref. M-368).*

• Vendo VIC-20 con Datassette, curso de introducción al Basic I con sus cintas, guía del usuario, libro Acceso Rápido al VIC-20, cartucho Sargon II Chess y varios programas de utilidades y juegos, todo por 30.000 ptas. Llamar al (93) 346 79 88. Miquel Farré. C/ Josep Soldevilla, 4 - 5º 2ª. Barcelona. *(Ref. M-369).*

• Vendo VIC-20, comprado en noviembre 1983, libro Curso de Introducción al Basic Parte I con cintas, cartucho juego Ajedrez Sargon II, y 2 cintas con 20 programas tecleados. Todo por 25.000 ptas. Mi dirección: Francisco Lordán Villanueva. C/ Urgel, 230 - Atico 3ª. Tel. 259 00 42. 08036 Barcelona. *(Ref. M-370).*

• Vendo, por cambio de ordenador, VIC-20, con ampliación de 8K y la guía del usuario del VIC-20, el libro Acceso Rápido al VIC-20 (castellano) y la introducción al Basic 1º y 2º tomos, todo por 30.000 ptas. Vendo calculadora programable en basic Casio PB-110 (sin estrenar) por 8.000 ptas. Regalo baterías de recambio. Las dos cosas sólo 35.000. Interesados llamar al (93) 422 68 42, o escribir a Manuel González Rey. C/ Dr. Martí Juliá, 193 - 3º 2ª. Hospitalet de Llobregat (Barcelona). *(Ref. M-371).*

• Vendo VIC-20 y Datassette por 26.500, memoria 64K RAM por 14.500, todo junto por 40.500. Llamar de 15 a 15,40 y de 21,30 a 10,30. Tel. 253 09 87. Barcelona. *(Ref. M-372).*

• Estoy interesado en adquirir la unidad de disco 3040 ó 4040 en buen estado. Dispuesto a intercambio de programas CBM 3032 ó 4032. Angel Fuentes Perille. C/ Guadalete, 11 - 3A. Cartagena. Tel. (968) 51 03 88. *(Ref. M-373).*

• Vendo ordenador VIC-20 en perfecto estado, junto con revistas, cartuchos de juegos, cintas, la 1ª parte del libro "Introducción al lenguaje de programación Basic y sus respectivas cintas para una mayor compresión del libro. Precio a convenir. Llamar al (93) 372 26 39, o escribir a Antonio Manuel Prieto Gómez, calle Doctor Manuel Riera, 69 - 1º 4ª. Esplugues de Llobregat (Barcelona). *(Ref. M-374).*

• Vendo C-64 más unidad de disco 1541 más cartucho expansión "The Idol" más monitor B y N con sonidos; más varios programas utilidades más libros sobre C-64 y programas en francés. Todo en perfecto estado. José A. Prieto Alegre. Norte, 11. Andrautx (Baleares). Tel. (971) 67 12 71. *(Ref. M-375).*

• Intercambio juegos para Commodore 64 sólo en Alicante. Asimismo cedería el juego Saucer Attack por joystick quick shot 2 (de disparo automático), o bien cedería a cambio del mismo, mi joystick quick shot 1 (en buen estado), pagando la diferencia. Guillermo Luyk. C/ Manuel Antón, 10 - 5º Dcha. 03004 Alicante. Tel. 20 91 03. *(Ref. M-376).*

• Tengo un Commodore 64 y desearía intercambiar programas en cinta para radioaficionados. José Luis Espinar EA7EBT. Apdo. 222 o C/Camilo José Cela, 2 - 2 G. Fuengirola (Málaga). Tel. (952) 47 10 76. *(Ref. M-377).*

• Cambio mi SX-64 por ordenador Commodore 64 más unidad de discos 1541, más programas por valor de 50.000 ptas, más joystick. Interesados llamar o escribir a: Luis Paus. Ctra. de Vic, 119 -5º 2 Manresa (Barcelona) Tel. 873 08 29. *(Ref. M-378).*

• Me gustaría recibir todo tipo de catálogos e información acerca del software que hay escrito para el Commodore 64. Mi dirección es Víctor J. Calvo Medina. C/ Beatas, 20 - 3º D. 29008 Málaga. *(Ref. M-379).*

• Intercambiaría programas, en cinta, para C-64. Escribir a: Carles Buchaca. C/ Sant Pere, 21 - 4º 2ª. Igualada (Barcelona). *(Ref. M-380).*

• Poseo un CBM-64, por ello me interesaría intercambiar programas de diversos tipos. Interesados escribir a Javier Salas Varela. C/ Velázquez, 1. 11010 Cádiz, o llamar al tel. (956) 25 50 64. *(Ref. M-381).*

• Intercambio juegos, programas y aplicaciones para VIC-20. Escribir a: Carlos Genescá. C/ Aribau, 302 - 1º 1ª. 08006 Barcelona. Tel. 201 31 63. *(Ref. M-382).*

CLUBS

• Club de amigos. Foto Estudio 2. Plaza de Sombrereros, 2. **Palma de Mallorca**. Tel.: 21 31 62. *(Ref. C-1).*

• Club de programación Alaiz. Tels.: 254 480-257 704. Pamplona. *(Ref. C-4)*

• Club de Commodore de Albacete. Fernando Martínez Guerrero. La Roda, 39, 5º D. Albacete. *(Ref. C-5)*

• Círculo de Durensan Vigués. Vigo. C/Venezuela, 48. Entrechan. Tels.: (986) 410 683/422 519. *(Ref. C-6)*

• Asociación Manchega de Usuarios de CBM-64 (A.M.U.C.-64). c/ Blasco de Garay, 10. Albacete. c/ La Roda, 39-5º D. c/ Octavio Cuartero, 15-6º D. *(Ref. C-13).*

• Para formación club nacional de CBM Series 2000 y 3000 exclusivamente, ponerse en contacto por carta con Artemio González Pérez. Fernando Arocena Quintero, 1. Teléf.: 22 42 44. 38009 Santa Cruz de Tenerife. *(Ref. C-31).*

• Ha nacido el club "Brian & Chip" para usuarios del C-64 y Spectrum 16K. Crearemos una revista propia. Escribir desde cualquier rincón de España. Francisco Reig López, C/Marina, 261, 3º 5.ª, 08025 Barcelona. *(Ref. C-33).*

• Desearía información sobre clubs en existencia y gente que quiera formar uno en Pontevedra de Commodore 64. Carlos Parada Gandos. C/Padre Fernando Olmedo, 4-4º B. 36002 Pontevedra. Tel. (986) 85 69 21. *(Ref. C-43).*

• Ha nacido el Club de Informática y videojuegos, el que nos escriba recibirá su carnet, con nº, intentaremos crear un presidente por ciudad (el 1º en escribir de cada una). Juan A. Manso. C/Illescas, 81. 28024 Madrid *(Ref. C-44)*

• Zaragoza. Club de amigos del VIC-20 y C-64 en ESPUMOSOS los sábados de 9,30 a 12,30. Daniel Fdez. de Velasco, C/Gran Vía, 39. Zaragoza. *(Ref. C-45).*

• Les ruego si puede ser que me pongan en contacto con algún usuario del VIC-20. Daniel Roig. C/Castilla, 27. Ibiza (Baleares) *(Ref. C-46).*

• Me gustaría contactar con alguien de **Figueras** o alrededores para intercambiar programas o formar un Club Commodore 64. Josep Teixidor Planas, Vilasacra, 2. El Far D, Emporda (Girona). Tel. 508649. *(Ref. C-47).*

• VIC-20 Desearía un intercambio de experiencias en cuanto a programación, aplicaciones al campo de la radioafición (intefaces) y programas diversos. EC 6 MF. Lorenzo Sabater Coll. C/Agua, 2. Esporlas, Mallorca, Baleares. *(Ref. C-48).*

• Estamos formando un club de usuarios del COMMODORE 64 en Las Palmas. Desearíamos contactar con chicos y chicas para intercambio de programas y experiencias. Interesados dirigirse a: César Montenegro Armas, C/Pío XII, 56 2-D. Tel. 24 60 61. *(Ref. C-49).*

• Desearía ponerme en contacto con usuarios del VIC-20 para intercambiar programas e información. Carles Balañá. C/Tívoli, nº 11, 5ºA. Réus (Tarragona). *(Ref. C-50).*

• Som dos nois de SABADELL que estem interessats en formar un club d'usuaris d'ordenadors COMMODORE i SINCLAIR. Els interassats o interassades, poden escriure a: Daniel Ribes Garolera, C/Via Aurelia nº 24, 2º 3º Sabadell. Tambe demanen si es posible informació sobre clubs ja existens. *(Ref. C-51).*

• Desearía contactar por correo con usuarios del VIC-20, tanto usuarios nacionales como extranjeros. Escribir a Mª Mercedes a la C/Marqués del Muni, 25 (Guía de Gran Canaria). *(Ref. C-52).*

• Se ha creado "COESPECTRUM 16", un nuevo Club, nacido para todos los usuarios de CBM 64, Spectrum 16K y Dragon 32 y 64. Pero para que todo ello se haga realidad, necesitamos colaboradores y muchas ideas. Se intercambiarán toda clase de programas o listados. Se creará una revista. Escribir a Avd. de la Aurora, 57, 12º A. (29006) Málaga, desde cualquier punto de España, se contestará o llamar a: Angel, Tel. 321799, o Emilio, Tel. 398169. *(Ref. C-53).*

• Desearía contactar con interesados para formar un club de usuarios en Vigo (Pontevedra) (o sus alrededores). A ser posible entre 12 y 16 años (aunque de otra edad me es prácticamente igual). Telf. 37 86 77. Preguntar por José Manuel Mera Alonso. C/ Buenos Aires, 2-2º C. Vigo (Pontevedra). *(Ref. C-54).*

• Club Commodore del Bager con una amplia programoteca (más de 250 programas) y gran cantidad de accesorios para CBM 64 y Vic-20 deseamos contactar con otros clubs y usuarios para intercambio. Dirigirse a: Félix Portabella. C/ Forn de Sta. Llucia, 1-2.ª. Manresa (Barcelona). Telf. 872 72 97 u Oscar 872 20 71. *(Ref. C-55).*

• Me gustaría contactar con usuarios de Commodore 64 de La Coruña para intercambiar programas y experiencia sin que nos cueste nada si se dan las condiciones apropiadas. Si estais interesados mis señas son: Gonzalo Montoto Veira. Avd. Sardiñeira, 5-2º. 15007 La Coruña. *(Ref. C-56).*

• Quisiera contactar con usuarios de Commodore-64, para formar un club aquí en Málaga. Podríamos compartir muchas experiencias, programas, etc. y salir ganando todos. Contactar con Víctor Calvo Medina. Telf. 21 44 09 de Málaga. *(Ref. C-57).*

• Agradecería recibir direcciones de commodorianos-64 residentes en Guipúzcoa en especial Irún, Fuenterrabía, Mondragón y Vergara. Manuel Prieto Ugarte. C/ Izagu, 5. Irún (Guipúzcoa). Telf. 62 47 38. *(Ref. C-58).*

• Me gustaría recibir información sobre clubs en Sevilla, como para poder intercambiar programas, etc. Pablo Jesús Astorga Lara. Residencia Conde de Bustillo, bloque 11-7º F. Telf. 63 52 37. Sevilla-5. *(Ref. C-59).*

• Si deseáis formar un club de usuarios de ordenadores VIC-20 en Córdoba, escribidme a: Francisco Jesús Gutiérrez Izquierdo. c/ Prolongación de Escañuela, 2. 14002-Córdoba. Telf. 25 08 73 (de 19 a 21 h.). *(Ref. C-60).*

• SPRITE. Club de usuarios de Commodore 64. Sin ánimo de lucro. Dirigirse a Pedro Javier Muñóz Pérez. c/ Zarandona, 48. Santiago de la Ribera (Murcia). *(Ref. C-61).*

• Desearía formar un club de amistad para toda España, para intercambiar programas para VIC-20, cambiar experiencias, sin gastarnos dinero. Daniel Roig Marchuet. c/ Castilla, 27 (Bajos). Ibiza (Baleares). *(Ref. C-62).*

• Deseo contactar con usuarios de Superbase-64 para intercambio de experiencias y/o programas. Josep Rovira i Sardá. c/ Dr. Escayola, 17-1º-2.ª. Telf. (93) 891 07 40. San Sadurni de Noya. *(Ref. C-63).*

• Desearía contactar con algún club de usuarios de C-64 en Málaga. Marcos García. Rueda. Telf. 27 11 36 *(Ref. C-64).*

• Quiero contactar con chico-chicas que empiezan a programar en código máquina. Intercambiaríamos experiencias, información, etc. Escribir a Robert Martín. Font, 21. Cassa de la Selva (Girona). Tengo un C-64. *(Ref. C-65).*

• Propongo crear una red de intercambio por teléfono mediante modem. Si alguien se lo desea autoconstruir, puedo facilitar esquema y conexionado. Interesa conseguir software para intercomunicación con modem. Llamar al (93) 333 50 80. Pedro Margalef. Cardenal Reig, 23 - 2º 1ª. 08028 Barcelona. *(Ref. C-66).*

• Apúntate al club "SPECDORE", para usuarios del Commodore 64 y Spectrum. Respuesta garantizada. Para inscribirse preguntar por Juanvi en el tel. (943) 45 10 09, o bien, escribir a C/ Prim, 37 - 2º 20006 San Sebastián (Guipúzcoa). *(Ref. C-67).*

• Estoy interesado en ponerme en contacto con usuarios del C-64 que posean algún programa de traducción de textos potente, a ser posible en disco, de inglés a español. Asimismo, quisiera conseguir las instrucciones del compilador Pascal 64 versión 3.0. Quien tenga información sobre esto escriba a: Mariano Prado Zorzano. C/ Albia de Castro, 3. 26003 Logroño. Tel. (941) 23 01 54. Estaría dispuesto a comprarlo o cambiarlo por programas, esto a convenir. *(Ref. C-68).*

• Estoy interesado en intercambiar programas, en especial educativos. También quiero entrar en contacto con profesores de EGB, BUP y FP que estén interesados en la programación, sobre todo si están llevando alguna experiencia en la escuela. Manuel del Arbol Sánchez. C/ Arrumbadores, 4 - 4º. Montilla (Córdoba). *(Ref. C-69).*

• Se ha formado el "SINCOM" Club de Spectrum y Commodore 64. Se creará una revista. Manda sobre con tu dirección debidamente franqueado para contestarte con la máxima rapidez. Escribid a SINCOM. Avda. Virgen de Guadalupe, 20 - 6 F. 10001 Cáceres. *(Ref. C-70).*

EXCURSION POR LA MICROPROGRAMACION

Rincón del Código Máquina

El cursillo como tal ha terminado, pero seguiremos viendo rutinas útiles de código máquina en este "Rincón del Código Máquina"; en él veremos varias rutinas que, combinadas en nuestros programas Basic, nos permitirán incrementar la velocidad de las funciones que son más lentas o simplemente no podemos realizar sin recurrir al código máquina.

En este número publicamos una rutina que permite copiar el BASIC de ROM a RAM; de este modo podremos alterar las de las funciones del intérprete a nuestro gusto. La rutina ocupa solamente las posiciones de memoria $C000 a $C016, ¡con sólo 22 bytes de programa podemos hacer en menos de un segundo lo que tardaría más de medio minuto en Basic!

Para copiar el intérprete de ROM a RAM basta con incorporar a nuestro programa una línea en que hagamos: SYS 49152.

Después de ejecutar la rutina, ya tenemos el basic en la RAM que se encuentra bajo la ROM, y si queremos, podemos hacer POKE 1,54 para cambiar la configuración de memoria con la que trabaja la CPU. Esta rutina podéis utilizarla en vuestros propios programas sin tener que teclear las siguientes.

El programa en código máquina que ocupa las direcciones de memoria $C017 a $C030 se puede utilizar también por sí solo, ya que no utiliza las restantes rutinas. Este programa permite añadir un nuevo comando a nuestro Basic: el PRINT AT. Aunque no lo añade realmente a la lista de comandos del intérprete, lo podemos simular con el comando:

Por Diego ROMERO



SYS 49175, columna, fila, "texto".

También se puede acortar el tamaño del programa si le asignamos el valor 49175 a una variable y ganamos cuatro bytes cada vez que debemos llamar a la rutina PRINT AT, esto se haría así:

```
10 P = 49175
.
.
1000 SYS P, columna, fila,
     "funciona"
.
.
1500 SYS P, columna, fila, a$
```

Supongo que este nuevo comando servirá de ayuda a muchos que están acostumbrados a utilizarlo en otros ordenadores y facilitará la tarea de conseguir una buena representación en pantalla para los programas.

A muchos les gustaría disponer de instrucciones GOTO y GOSUB calculadas, pues están acostumbrados a utilizarlas con otras versiones de Basic y les parece difícil de utilización masiva de comandos ON... GOTO y ON... GOSUB. Para éstos va destinada la rutina que ocupa las direcciones $C031 a $C048. Esta rutina necesita que esté en su lugar la de copia de ROM a RAM (la primera rutina), ya que debe copiar el intérprete Basic y cambiar una instrucción en la rutina que se encarga de los comandos GOTO y GOSUB para que al ejecutarlos pase por la nuestra (NUEGOT).

Para activar los GOTO y GOSUB calculados es necesario teclear el comando:

SYS 49207

y podéis comprobar que todo va bien con un programa como el siguiente:

```
10 L=40 : GOTO L
20 PRINT "ESTO NO
   FUNCIONA"
30 END
40 PRINT "ESTO FUNCIONA
   BIEN"
```

En lugar de hacer GOTO L se pueden teclear fórmulas complejas detrás del GOTO, por ejemplo GOTO 2*(5*(L/2))/5.

Otro comando sobre el que hemos recibido varias peticiones por carta es el RESTORE N, de modo que si debemos repetir varias veces una secuencia de datos en un programa podamos utilizarlo sin necesidad de leer todas las líneas DATA hasta encontrar la línea que nos interesa.

Para añadir este comando al intérprete Basic, necesitamos utilizar la rutina que ocupa las posiciones de memoria $C049 a $C071. Para que este programa funcione es necesario que estén en sus respectivas posiciones de memoria las rutinas de los GOTO y la de copia de memoria, ya que son llamadas tanto al activarla como al ejecutarse.

Para añadir este comando sólo es necesario teclear SYS 49252.

No es necesario haber activado la rutina anterior, ya que este

EXCURSION POR LA MICROPROGRAMACION

último SYS la activará automáticamente, añadiendo los comandos GOTO N, GOSUB N y RESTORE N.

Para terminar por este mes, la rutina que ocupa las posiciones $C072 a $C08C está destinada a 'mirar' el contenido de la RAM que se encuentra escondida debajo de la ROM del KERNAL y del BASIC. Inicialmente fue pensada para ser utilizada solamente como PRINT PEEK (DIRECCION), pero luego le quité la llamada a la rutina PRINT, y añadí las instrucciones que ocupan las direcciones $C08D a $C0A2.

Esta última rutina permite asignar el valor del 'PEEK' a una variable de tipo numérico. Para hacer un PEEK bajo la ROM debemos teclear la siguiente línea en Basic:

SYS 49293, variable, dirección.

Por tanto si siquiésemos saber el contenido de la dirección 65000 de RAM, podríamos hacerlo:

10 SYS 49293,A,65000
20 PRINT A

La dirección puede ser también otra variable, pues la rutina del PEEK bajo ROM se encarga de 'evaluar' la dirección por medio de la llamada a la rutina EVAARG del intérprete Basic ($C075 JSR EVAARG).

****Una observación importante: la variable que se utiliza para ver el contenido de la RAM oculta debe ser del tipo de coma flotante, ya que la rutina de código máquina no comprueba si lo es o no, y en caso de utilizar una variable entera (por ejemplo A%)

> **Supongo que este nuevo comando servirá de ayuda a muchos que están acostumbrados a utilizarlo en otros ordenadores y facilitará la tarea de conseguir una buena representación en pantalla para los programas.**

dará valores que nada tienen que ver con la realidad.

Espero que sean útiles para todos las rutinas que he recopilado, y sobre todo la última para aquellos que, pese a tener 38K RAM libres, les sigue pareciendo poco y quieren disponer de la RAM que les han 'robado' el intérprete y el kernal. Otra aplicación de esta rutina sería para proteger vuestros programas, ya que podríais esconder datos bajo la ROM y hacer que no funcionaran los programas si los datos no fuesen correctos.

Además de las posibles aplicaciones que les podáis dar a estas rutinas, creo que la principal es la de aprender observando el modo de llamar las rutinas del intérprete, y podréis intentar utilizarlas en vuestros programas. Por ejemplo, la rutina BUSCOM ($AEFD) sirve para comprobar si el siguiente carácter del programa basic es una coma, y en caso contrario da un "SYNTAX ERROR".

La rutina EVAARG ($AD8A) evalúa el argumento de una función o comando Basic, y el resultado lo guarda en el acumulador de coma flotante número uno.

La rutina FACADR ($B7F7) sirve para convertir un número entero situado en el acumulador número uno a formato binario de 16 bits, ya que aunque trabajemos con números enteros en el programa basic, el ordenador los trata como números de coma flotante para todos sus cálculos aritméticos.

La rutina $B08B sirve para encontrar la dirección de memoria que ocupa una variable Basic, y devuelve la dirección física de la variable en los registros A e Y de la CPU.

La rutina $B391 pasa al acumulador número uno el número binario contenido en los registros A e Y.

La rutina $BD0 pasa los datos del acumulador número uno a la variable señalada por el contenido de las posiciones $49 y $4A.

Para aquellos que quieran cargar estas rutinas en la memoria del ordenador y no dispongan de un ensamblador, hemos preparado un programa en basic que contiene las rutinas de código máquina en forma de sentencias DATA con un bucle que las va leyendo y colocando en sus posiciones respectivas; es el listado 2.

Para facilitar la detección de errores al teclearlo he añadido también la suma de control del programa (ver número 10, página 16).

PROGRAMAS UTILITARIOS

```
                 ; ESTE ES UN CONJUNTO DE RUTINAS
                 ; CON DIVERSAS FUNCIONES UTILES
                 ; PARA INCORPORARLAS A NUESTROS
                 ;     PROPIOS PROGRAMAS.
                 ;   (C) DIEGO ROMERO 1985
                 ; COMMODORE WORLD-C.W.SPAIN
                 ;
                 BASIC   =   $A000
C000             VECTOR  =   $FE     ; VECTOR LIBRE
C000             VRESTO  =   $A024   ; VECTOR DE RESTORE
C000             RUTGOT  =   $A8A0   ; RUTINA GOTO
C000             PARIZQ  =   $AEF7   ; BUSCA (
C000             PARDER  =   $AEFA   ; BUSCA )
C000             BUSCOM  =   $AEFD   ; BUSCA COMA
C000             PLOT    =   $FFF0   ; LEE/FIJA POSICION
C000             PRINT   =   $AAA0   ; RUTINA PRINT
C000
```

EXCURSION POR LA MICROPROGRAMACION

```
1170:  C000          LEEBYT   =    $B79B      ; LEE ARGUMENTO (0-255)
1180:  C000          EVAARG   =    $AD8A      ; EVALUA ARGUMENTO
1190:  C000          FACADR   =    $B7F7      ; PASA FAC A ADDRESS
1195:  C000          CALADR   =    $A613      ; CALCULA DIRECCION DE LINEA
1197:  C000          ILEGAL   =    $B248      ; ILEGAL QUANTITY
1200:  C000                   *=   $C000
                     ;
                     ; RUTINA DE COPIA DEL BASIC
                     ;
1210:  C000 A2 A0    RAMBAS   LDX  #>BASIC    ; DIRECCION INICIAL
1220:  C002 A0 00             LDY  #<BASIC    ; DE COPIA ROM-RAM
1230:  C004 86 FF             STX  VECTOR+1
1240:  C006 84 FE             STY  VECTOR
1250:  C008 A2 20             LDX  #$20       ; PAGINAS A COPIAR
1260:  C00A B1 FE    COPIA    LDA  (VECTOR),Y
1270:  C00C 91 FE             STA  (VECTOR),Y ; COPIA UN BYTE
1280:  C00E C8                INY
1290:  C00F D0 F9             BNE  COPIA      ; SIGUIENTE BYTE
1300:  C011 E6 FF             INC  VECTOR+1
1310:  C013 CA                DEX             ; CUENTA PAGINAS
1320:  C014 D0 F4             BNE  COPIA      ; SIGUIENTE PAGINA
1330:  C016 60                RTS
                     ;
                     ; RUTINA PRINT-AT
                     ;
1340:  C017 20 FD AE          JSR  BUSCOM
1350:  C01A 20 9E B7          JSR  LEEBYT+3   ; NUMERO COLUMNA
1360:  C01D 86 FE             STX  VECTOR     ; GUARDA EL VALOR
1380:  C01F 20 FD AE          JSR  BUSCOM
1390:  C022 20 9E B7          JSR  LEEBYT+3   ; NUMERO DE LINEA
1400:  C025 A4 FE             LDY  VECTOR     ; RECUPERA COLUMNA
1410:  C027 18                CLC
1420:  C028 20 F0 FF          JSR  PLOT       ; COLOCA EL CURSOR
1430:  C02B 20 FD AE          JSR  BUSCOM     ; BUSCA COMA
1440:  C02E 4C A0 AA          JMP  PRINT      ; VA A PRINT NORMAL
                     ;
                     ; RUTINA GOTO/GOSUB CALCULADOS
                     ;
1480:  C031 20 8A AD NUEGOT   JSR  EVAARG     ; LEE ARGUMENTO
1490:  C034 4C F7 B7          JMP  FACADR     ; LO CONVIERTE EN DIRECCION
1500:  C037 20 00 C0 CAMBIA   JSR  RAMBAS     ; PASA BASIC A RAM
1505:  C03A A9 C0             LDA  #>NUEGOT
1510:  C03C A2 31             LDX  #<NUEGOT
1520:  C03E 8D A2 A8          STA  RUTGOT+2   ; PONE NUEVO VECTOR
1530:  C041 8E A1 A8          STX  RUTGOT+1   ; EN RUTINA GOTO
1550:  C044 A9 36             LDA  #54
1560:  C046 85 01             STA  $01        ; CAMBIA A RAM
1570:  C048 60                RTS
                     ;
                     ; RUTINA PARA RESTORE NN
                     ;
1610:  C049 EA       NRESTO   NOP
1615:  C04A F0 15             BEQ  ARESTO     ; NUEVO RESTORE
1620:  C04C 20 31 C0          JSR  NUEGOT     ; CALCULA LINEA DESTINO
1630:  C04F 20 13 A6          JSR  CALADR     ; LA PASA A DIRECCION
1640:  C052 38                SEC
1650:  C053 A5 5F             LDA  $5F        ; LSB DE DIRECCION
1660:  C055 E9 01             SBC  #1         ; LE RESTA 1
1670:  C057 A4 60             LDY  $60        ; MSB DE DIRECCION
1680:  C059 B0 01             BCS  ETIQU1     ; NO HAY ARRASTRE
1690:  C05B 88                DEY
1700:  C05C 85 41    ETIQU1   STA  $41        ; PUNTERO DE DATAS
1710:  C05E 84 42             STY  $42
1720:  C060 60                RTS
1730:  C061 4C 1D A8 ARESTO   JMP  $A81D      ; ANTIGUO RESTORE
1740:  C064 20 37 C0 ACTIVA   JSR  CAMBIA
1750:  C067 A9 C0             LDA  #>NRESTO   ; MSB DEL NUEVO RESTORE
1760:  C069 A2 49             LDX  #<NRESTO   ; LSB
1770:  C06B 8D 25 A0          STA  VRESTO+1
1780:  C06E 8E 24 A0          STX  VRESTO
```

EXCURSION POR LA MICROPROGRAMACION

```
                                RTS
C071 60
                        ;
                        ; RUTINA PEEK BAJO ROM
                        ;
C072 20 FD AE PEKROM      JSR  BUSCOM   ; BUSCA COMA
C075 20 8A AD             JSR  EVAARG   ; LEE LA DIRECCION
C078 20 F7 B7             JSR  FACADR   ; LA PASA A 16 BITS
C07B 78                   SEI           ; INHIBE INTERRUPCIONES
C07C A5 01                LDA  $1       ; CONFIGURACION ACTUAL
C07E 48                   PHA           ; LA GUARDA EN EL STACK
C07F 29 FD                AND  #$FD
C081 85 01                STA  $1       ; QUITA ROM
C083 A2 00                LDX  #0
C085 A1 14                LDA  ($14,X)  ; LEE EL BYTE DE RAM
C087 A8                   TAY           ; LO PASA AL REGISTRO Y
C088 68                   PLA           ; RECUPERA CONFIGURACION
C089 85 01                STA  $1
C08B 58                   CLI           ; REACTIVA INTERRUPCIONES
C08C 60                   RTS
                        ;
                        ; RUTINA VARIABLE=PEEK(ROM)
                        ;
C08D 20 FD AE             JSR  BUSCOM   ; BUSCA COMA
C090 20 8B B0             JSR  $B08B    ; BUSCA LA VARIBLE
C093 85 49                STA  $49      ; GUARDA LA DIRECCION
C095 84 4A                STY  $4A      ; DE LA VARIABLE
C097 20 72 C0             JSR  PEKROM   ; PEEK BAJO ROM
C09A A9 00                LDA  #0
C09C 20 91 B3             JSR  $B391    ; 16 BITS AL ACUM 1
C09F 20 D0 BB             JSR  $BBD0    ; FAC A VARIABLE
C0A2 60                   RTS
```

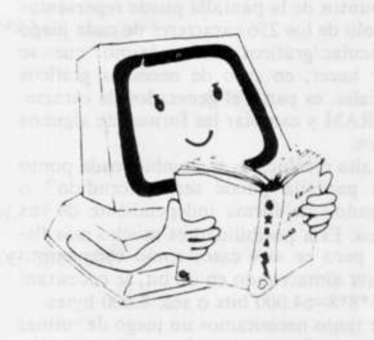
PROGRAMAS UTILITARIOS

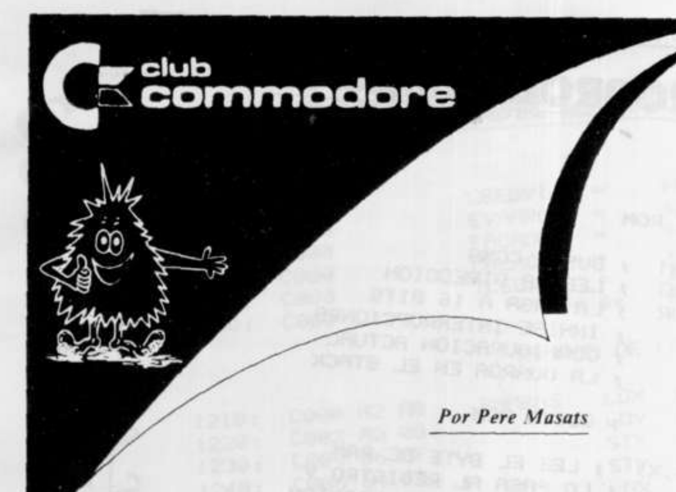
```
10 I=49152
20 READ A:IF A=256 THEN 40
30 POKE I,A:I=I+1:S=S+A:GOTO 20
40 IF S<>21206THENPRINT"ERROR EN DATOS"
50 END
49152 DATA 162,160,160,0,134,255,132,254
49160 DATA 162,32,177,254,145,254,200,208
49168 DATA 249,230,255,202,208,244,96,32
49176 DATA 253,174,32,158,183,134,254,32
49184 DATA 253,174,32,158,183,164,254,24
49192 DATA 32,240,255,32,253,174,76,160
49200 DATA 170,32,138,173,76,247,183,32
49208 DATA 0,192,169,192,162,49,141,162
49216 DATA 168,142,161,168,169,54,133,1
49224 DATA 96,234,240,21,32,49,192,32
49232 DATA 19,166,56,165,95,233,1,164
49240 DATA 96,176,1,136,133,65,132,66
49248 DATA 96,76,29,168,32,55,192,169
49256 DATA 192,162,73,141,37,160,142,36
49264 DATA 160,96,32,253,174,32,138,173
49272 DATA 32,247,183,120,165,1,72,41
49280 DATA 253,133,1,162,0,161,20,168
49288 DATA 104,133,1,88,96,32,253,174
49296 DATA 32,139,176,133,73,132,74,32
49304 DATA 114,192,169,0,32,145,179,32
49312 DATA 208,187,96,256
```

PROGRAMAS UTILITARIOS

SUMA DE CONTROL DEL CARGADOR BASIC

10	0	20	168	30	136
40	185	50	128	49152	51
49160	112	49168	67	49176	68
49184	72	49192	13	49200	22
49208	20	49216	21	49224	171
49232	183	49240	179	49248	200
49256	13	49264	20	49272	163
49280	155	49288	174	49296	222
49304	222	49312	109		





Por Pere Masats

Programa para realizar gráficos de alta resolución en el C-64

Para aquellos que estén interesados en la alta resolución del COMMODORE 64, en este artículo vamos a suministrar un utilitario en código/máquina que permite realizar dibujos en alta resolución desde BASIC.

Una de las caracteristicas más interesantes del COMMODORE 64 es la de poder realizar dibujos en pantalla en el modo llamado de ALTA RESOLUCION. Esto quiere decir que se puede dibujar con una resolución (número de puntos) 300×200 en dos colores. Desafortunadamente el intérprete de BASIC incorporado en el equipo no posee instrucciones específicas para utilizar esta capacidad de dibujo en alta resolución, siendo necesario recurrir a rutinas con POKEs y PEEKs en cantidad suficiente para marear a cualquiera. A continuación presentaremos una serie de rutinas en código/máquina para realizar dibujos en alta resolución, no se trata de un programa que realice gráficos, es en realidad un lenguaje específico para dibujar en alta resolución.

En modo alfanumérico (el normal) se almacena en un espacio de memoria de 1000 bytes el contenido de cada carácter de la pantalla. Como la capacidad es de 25 filas de 40 caracteres, se necesita un total de 1000 bytes, este espacio —como Vd. ya debe saber— normalmente está situado entre 1024 y 2023 ($0400-$07E7). En este modo cada matriz de 8×8 puntos de la pantalla puede representar uno solo de los 256 caracteres de cada juego minúsculas/gráficos. Lo máximo que se puede hacer, en caso de necesitar gráficos especiales, es pasar el generador de caracteres a RAM y cambiar las formas de algunos de ellos.

En alta resolución, en cambio, cada punto de la pantalla puede ser "encendido" o "apagado" de forma independiente de sus vecinos. Esta posibilidad es mucho más flexible, pero en este caso, como cada punto debe ser almacenado en un bit, se necesitan: 40*25*8*8=64.000 bits o sea: 8.000 bytes.

Por tanto necesitamos un juego de rutinas que siendo lo más rápidas posible puedan cumplir las siguientes funciones:

Pasar de modo alfanumérico normal a alta resolución.

Limpiar la pantalla de alta resolución en un momento dado.

Dado un par de coordenadas X, Y encender o apagar el punto correspondiente.

Dados dos pares de coordenadas, dibujar una recta entre ellos.

Hacer que estas rutinas sean fácilmente utilizables desde BASIC.

Y, por último, proporcionar un fácil retorno al funcionamiento normal del BASIC, incluso si se ha cometido un error.

En la figura 1 se da el listado de un programa en código/máquina que realiza estas funciones. En la figura 2 se lista el equivalente en BASIC de dicho programa en forma de sentencias DATA para aquellos usuarios que no dispongan de un ASSEMBLER o un MONITOR DE CODIGO MAQUINA, por último, en la figura 3, se da el listado de un programa de demostración en BASIC que utiliza los nuevos comandos especiales para alta resolución. En las figuras siguientes se dan muestras del aspecto de la pantalla durante el funcionamiento del programa de demostración.

En el listado de la figura 3 se puede observar el método utilizado para incorporar en los programas en BASIC los nuevos comandos de alta resolución. Se trata de realizar SYS a posiciones específicas seguidos por los parámetros correspondientes. Este método de "extender" el BASIC puede parecer a primera vista algo tosco, pero tiene la ventaja de no requerir manipulaciones complicadas en el intérprete que redundarían en una pérdida de velocidad.

Los comandos implementados son:

SYS (IN).—Sin necesidad de parámetros, inicializa el programa de alta resolución. Debe ser utilizado antes de cualquier otra instrucción de dibujo en alta resolución.

SYS (RS).—Esta instrucción finaliza el uso del lenguaje gráfico, devuelve la pantalla al uso normal alfanumérico. Debe emplearse al final de los programas para obtener el mensaje READY.

SYS (CL).—Este comando borra la pantalla en alta resolución. Como los anteriores, no requiere parámetros.

SYS (PX), X, Y.—Esta instrucción enciende (o dibuja) un punto en las coordenadas especificadas por los valores de X e Y. Cuando X e Y son iguales a cero, el punto es el de la esquina inferior izquierda de la pantalla.

SYS (MV) X, Y.—Este comando sitúa el cursor de punto en las coordenadas especificadas sin efectúar ningún dibujo.

SYS (DR), X, Y.—Dibuja una línea entre la posición actual del cursor de punto y las coordenadas especificadas.

POKE (CR+1), COLOR: SYS (CR).—Mediante estos comandos se establecen los colores de fondo y del dibujo. En la página 61 del MANUAL DEL USUARIO DEL COMMODORE 64 se da una tabla de los números correspondientes a cada uno de los 16 colores, el valor de COLOR se obtiene de la siguiente manera:

COLOR=16 * (NUMERO DEL COLOR DEL DIBUJO) + (NUMERO DEL COLOR DEL FONDO).

Descripción del Programa

A continuación se describe el programa que permite realizar gráficos de alta resolución, véase la figura 1.

En las líneas 41-107 se establecen las variables, constantes y origen del programa.

En las direcciones $6000 a $600E se implementa una tabla de saltos para cada una de las instrucciones.

La rutina HRMOVE ($600F-$6088) calcula, dadas las coordenadas, el byte y el bit dentro de él que corresponden a estas coordenadas. En esta y otras rutinas se puede entrar en otro lugar si los valores ya están establecidos, es decir, si se ha efectuado un salto desde código/máquina en vez de desde un programa en BASIC, en este caso esta rutina empieza en $602D.

VECPLT ($6089-$61C5) es la rutina principal de dibujo de una recta, contiene también la subrutina para mover el cursor sin dibujar.

RNGCHK ($61C6-$61E6) chequea los parámetros que enviamos con los comandos y genera un mensaje de error ?ILLEGAL QUANTITY en caso necesario.

Entre las direcciones $61ED y $625A se listan las rutinas para realizar los movimientos elementales.

De $625B a $6276 es la rutina que realiza el borrado de la pantalla en alta resolución.

SETCOL ($6277 - $628F) es la rutina que realiza la operación de llenar la memoria de color con la combinación deseada.

HRINIT ($6290 - $62BC) es la rutina que inicializa el modo gráfico en alta resolución, para ello primero modifica los registros del chip de vídeo, luego establece los colores del dibujo (llama a SETCOL), limpia la pantalla (llama a CLRHR) y modifica el vector de errores del BASIC para poder manejar los específicos de alta resolución.

HRREST ($62BD - $62E1) realiza la operación contraria: restablece los parámetros de vídeo para modo alfanumérico y restablece el vector de errores.

GETVAL ($62E2-$62EB) es la rutina que permite al lenguaje gráfico leer y evaluar los parámetros que siguen a las instrucciones. Para ello busca la coma (que actúa como delimitador de parámetros), evalúa la expresión que la sigue y el resultado en coma flotante que se obtiene lo convierte a entero, almacenándolo en los registros A e Y.

SETPIX ($62EC-$62F8) efectúa el encendido del punto una vez establecidos el byte y el bit por HRMOVE.

ABRT ($62F9-$6308) es la rutina que recibe el control en caso de error.

MOVTAB y MSKTB son tablas para uso del programa.

```
LIN    DIR. HEXADEC. ETIQUETA COD OPER      ;COM. PRG: "ALTA RESOLUCION "
2
                    ;************************************
                    ;*                                  *
                    ;*  PAQUETE PARA GRAFICOS EN ALTA   *
                    ;*                                  *
                    ;*  RESOLUCION EN EL COMMODORE 64   *
                    ;*                                  *
                    ;* MICROELECTRONICA Y CONTROL S.A.  *
                    ;*                                  *
                    ;************************************
                    ;
41:     6000        ORIGIN    =    $6000
                    ;
                    ;****** RUTINAS DEL SISTEMA *******
                    ;
47:     6000        ERROR     =    $A437   ;EXHIBIR MENSAJE DE ERROR.
48:     6000        EVAEXP    =    $AD9E   ;EVALUAR EXPRESION.
49:     6000        CHKCOM    =    $AEFD   ;BUSCAR LA COMA.
50:     6000        FLTFIX    =    $B1AA   ;CONVERTIR A FIJO EN Y (BAJ.) Y A
(ALT.).
                    ;
                    ;
                    ;*********** VECTORES ***********
                    ;
                    ;
56:     6000        ERRVEC    =    $0300   ;RUTINA DE ERROR.
57:     6000        WARMV     =    $0302   ;INICIO DE BASIC (WARM).
                    ;
                    ;******* VARIABLES DE VIDEO *******
                    ;
                    ;
63:     6000        VIC       =    $D000    ;DIRECCION DEL CHIP DE VIDEO.
64:     6000        HRCTRL    =    VIC+17   ;CONTROL DE MODO.
65:     6000        HRMREG    =    VIC+24   ;CONTROL DE MEMORIA.
                    ;
67:     6000        SCREEN    =    $0400    ;MEM. DE PANTALLA.
68:     6000        SCREEND   =    SCREEN+999 ;ULTIMA DIRECCION DE LA PANTALLA.
69:     6000        BASE      =    $2000    ;INICIO DE ALTA RES.
70:     6000        HRLAST    =    BASE+7999 ;ULTIMA POSICION.
71:     6000        RAM       =    $033C    ;BUFFER DEL CASSETTE.
                    ;
                    ;********* PAGINA CERO ***********
                    ;
                    ;
77:     6000        BYT       =    $FD      ;PUNTERO DE BYTE.
                    ;
                    ;
80:     033C                  *=   RAM
                    ;
                    ;
83:     033E        X1        *=   *+2      ;COORDENADA X (0 - 319).
84:     0340        X2        *=   *+2
85:     0342        Y1        *=   *+2      ;COORDENADA Y (0 - 199).
86:     0344        Y2        *=   *+2
87:     0345        BITNO     *=   *+1      ;BIT DEL BYTE.
88:     0347        DELTX     *=   *+2      ;X2-X1.
89:     0349        DELTY     *=   *+2      ;Y2-Y1.
90:     034B        E         *=   *+2
91:     034D        T         *=   *+2
92:     034F        C         *=   *+2
93:     0350        I         *=   *+1      ;PUNTERO DE DIRECCION.
94:     0352        TEMP      *=   *+2
95:     0354        ERVEC     *=   *+2      ;CONTIENE EL VECTOR DE ERROR.
                    ;
                    ;********* CONSTANTES ************
                    ;
101:    0354        XMAX      =    320
102:    0354        YMAX      =    200
103:    0354        COLS      =    40       ;NUM. DE COLUMNAS POR FILA.
104:    0354        COLOR     =    $50      ;COLOR DEL FONDO/TRAZO (VERDE OBRE
```

```
LIN    DIR. HEXADEC. ETIQUETA COD OPER        ;COM. PRG: "ALTA RESOLUCION "
NEGRO).
                                  ;
                                  ;
       6000                       *=   ORIGIN
                                  ;
                                  ;
                                  ;TABLA DE SALTOS.
                                  ;
       6000 4C 90 62 JINIT    JMP  HRINIT   ;INICIALIZAR.
       6003 4C BD 62 JREST    JMP  HRREST   ;VOLVER A BASIC.
       6006 4C 5B 62 JCLR     JMP  CLRHR    ;BORRAR PANTALLA.
       6009 4C 89 60 JDRAW    JMP  VECPLT   ;DIBUJAR RECTA.
       600C 4C EC 62 JSETPX   JMP  SETPIX   ;ENCENDER PUNTO (PIXEL).
                                  ;
                                  ;
                                  ;RUTINA HRADDR - DADAS COORD X ( 2 BYTES)
                                  ;Y COORD Y ( 1 BYTE )
                                  ;CALCULAR LA DIRECCION DEL BYTE Y EL
                                  ;BIT DENTRO DE EL (BITNO).
                                  ;
                                  ;ALTERA X, DEJA Y=0.
                                  ;
                                  ;SALTAR AQUI DESDE BASIC.
                                  ;
       600F 20 E2 62 HRMOVE   JSR  GETVAL   ;OBTENER X1.
       6012 8C 3C 03          STY  X1
       6015 8C 3E 03          STY  X2       ;PARA RNGCHK.
       6018 8D 3D 03          STA  X1+1
       601B 8D 3F 03          STA  X2+1
       601E 20 E2 62          JSR  GETVAL   ;OBTENER Y1.
       6021 8C 40 03          STY  Y1
       6024 8C 42 03          STY  Y2
       6027 8D 43 03          STA  Y2+1
       602A 20 C6 61          JSR  RNGCHK
                                  ;
                                  ;SALTAR AQUI SI X1, Y1 TIENEN VALOR.
                                  ;
       602D A9 00    HRADDR   LDA  #0       ;BYTE ALTO A CERO.
       602F 85 FE             STA  BYT+1
       6031 38                SEC           ;CALC. 199-Y1
       6032 A9 C7             LDA  #YMAX-1
       6034 ED 40 03          SBC  Y1
       6037 48                PHA           ;GUARDAR RESULTADO EN STACK.
       6038 29 F8             AND  #$F8     ;CALC. NUM. FILA.
       603A 0A                ASL  A        ;MULT. POR 2.
       603B 26 FE             ROL  BYT+1
       603D 0A                ASL  A        ;MULT. POR 4.
       603E 26 FE             ROL  BYT+1
       6040 0A                ASL  A        ;MULT. POR 8.
       6041 26 FE             ROL  BYT+1
       6043 48                PHA           ;GUARDAR EN STACK.
       6044 8D 50 03          STA  TEMP     ;Y EN TEMP.
       6047 A5 FE             LDA  BYT+1
       6049 8D 51 03          STA  TEMP+1   ;EN TEMP 8*Y
       604C 68                PLA           ;RESTAURAR A
       604D 0A                ASL  A        ;MULT. POR 16
       604E 26 FE             ROL  BYT+1
       6050 0A                ASL  A        ;MULT. POR 32
       6051 26 FE             ROL  BYT+1    ;(CARRY AUN 0)
       6053 6D 50 03          ADC  TEMP     ;CALC. 32+8 = 40
       6056 85 FD             STA  BYT      ;EN BYT
       6058 A5 FE             LDA  BYT+1
       605A 6D 51 03          ADC  TEMP+1
       605D 85 FE             STA  BYT+1
       605F AD 3C 03          LDA  X1       ;SUM CAR.
       6062 29 F8             AND  #$F8
       6064 65 FD             ADC  BYT
       6066 85 FD             STA  BYT
       6068 AD 3D 03          LDA  X1+1
       606B 65 FE             ADC  BYT+1
       606D 85 FE             STA  BYT+1    ;(CARRY AUN 0)
       606F 68                PLA           ;SUM LINEA
       6070 29 07             AND  #7       ;ENMASCARANDO BITS ALTOS
       6072 65 FD             ADC  BYT
       6074 85 FD             STA  BYT
       6076 A5 FE             LDA  BYT+1    ;ACABAR SUMANDO BASE
       6078 69 20             ADC  #>BASE
       607A 85 FE             STA  BYT+1
```

```
DIR. HEXADEC. ETIQUETA COD OPER     ;COM. PRG: "ALTA RESOLUCION "
607C AD 3C 03            LDA  X1        ;CALC. BITNO
607F 29 07               AND  #7        ;SON BITS BAJOS
6081 AA                  TAX            ;INDICE DE LA TABLA
6082 BD 29 63            LDA  MSKTB,X
6085 8D 44 03            STA  BITNO
6088 60                  RTS            ;BYT Y BITNO CALCULADOS
                ;
                ;********** FASTPLOT *************
                ;
                ;SUBRUTINA GRAFICA PARA DIBUJAR UNA RECTA
                ;EN LA MEMORIA DE ALTA RESOLUCION
                ;
                ;SE ENTRA CON VALORES PARA X1,Y1 Y X2,Y2
                ;Y EL PRIMER PUNTO ENCENDIDO, ESTO ES
                ;BYT, BYT+1, Y BITNO ESTAN CALCULADOS
                ;POR LLAMADA A LA RUTINA PIXADR
                ;
                ;DEJA Y=0, X ALTERADO Y TESTEA EL REBOSAMIENTO
                ;
                ;********* VECPLT ************
                ;
                ;SALTAR AQUI DESDE BASIC
                ;
6089 20 E2 62 VECPLT     JSR  GETVAL    ;OBTENER COORD. X
608C 8C 3E 03            STY  X2
608F 8D 3F 03            STA  X2+1
6092 20 E2 62            JSR  GETVAL    ;OBTENER COORDD Y
6095 8C 42 03            STY  Y2
6098 8D 43 03            STA  Y2+1
                ;
                ;SALTAR AQUI SI X2,Y2 ESTABLECIDOS
                ;
609B 20 C6 61 VECPL1     JSR  RNGCHK    ;VER. REBOS. DE X2,Y2
609E 38                  SEC            ;FORMAR DELTX (CON SIGNO)
609F AD 3E 03            LDA  X2
60A2 ED 3C 03            SBC  X1
60A5 8D 45 03            STA  DELTX
60A8 AD 3F 03            LDA  X2+1
60AB ED 3D 03            SBC  X1+1
60AE 8D 46 03            STA  DELTX+1
60B1 38                  SEC            ;FORMAR DELTY (CON SIGNO)
60B2 AD 42 03            LDA  Y2
60B5 ED 40 03            SBC  Y1
60B8 8D 47 03            STA  DELTY
60BB AD 43 03            LDA  Y2+1
60BE ED 41 03            SBC  Y1+1
60C1 8D 48 03            STA  DELTY+1
60C4 AD 3E 03            LDA  X2
60C7 8D 3C 03            STA  X1
60CA AD 3F 03            LDA  X2+1
60CD 8D 3D 03            STA  X1+1
60D0 AD 42 03            LDA  Y2
60D3 8D 40 03            STA  Y1
60D6 AD 43 03            LDA  Y2+1
60D9 8D 41 03            STA  Y1+1
                ;
                ;
                ;AHORA TENEMOS DELTX,DELTY (CON SIGNO)
                ;
                ;**** MOVE ****
                ;
                ;DADOS DELX,DELTY
                ;DIBUJAR/MOVER PARA LA MEJOR RECTA
                ;
60DC A9 00    MOVE       LDA  #0        ;DETERMINAR EL OCTANTE
60DE 8D 4F 03            STA  I
60E1 2C 46 03            BIT  DELTX+1 ;VER SI DELTX < 0
60E4 10 17               BPL  MV1
60E6 AD 45 03            LDA  DELTX     ;CAMBIAR SIGNO
60E9 20 E7 61            JSR  COMPL
60EC 8D 45 03            STA  DELTX
60EF AD 46 03            LDA  DELTX+1
60F2 20 E8 61            JSR  COMPH
60F5 8D 46 03            STA  DELTX+1
60F8 A9 02               LDA  #2
60FA 8D 4F 03            STA  I
60FD 2C 48 03 MV1        BIT  DELTY+1 ;VER SI DELTY < 0
6100 10 1B               BPL  MV2
```

```
DIR. HEXADEC. ETIQUETA COD OPER     ;COM. PRG: "ALTA RESOLUCION "

6102 AD 47 03           LDA  DELTY
6105 20 E7 61           JSR  COMPL
6108 8D 47 03           STA  DELTY
610B AD 48 03           LDA  DELTY+1
610E 20 E8 61           JSR  COMPH
6111 8D 48 03           STA  DELTY+1
6114 18                 CLC
6115 AD 4F 03           LDA  I
6118 69 04              ADC  #4
611A 8D 4F 03           STA  I
611D AE 45 03 MV2       LDX  DELTX     ;VER DELTX-DELTY
6120 EC 47 03           CPX  DELTY     ;CARRY 1 PARA BYTE BAJO
6123 AD 46 03           LDA  DELTX+1    ;BYTE ALTO
6126 A8                 TAY            ;Y=DELTX
6127 ED 48 03           SBC  DELTY+1
612A 10 1B              BPL  MV3
612C AD 47 03           LDA  DELTY     ;INTERCAMBIAR DELTX,Y
612F 8D 45 03           STA  DELTX
6132 AD 48 03           LDA  DELTY+1
6135 8D 46 03           STA  DELTX+1
6138 8E 47 03           STX  DELTY
613B 8C 48 03           STY  DELTY+1
613E 18                 CLC
613F AD 4F 03           LDA  I
6142 69 08              ADC  #8
6144 8D 4F 03           STA  I
6147 AD 45 03 MV3       LDA  DELTX     ;CALC. E=-DELTX/2
614A 20 E7 61           JSR  COMPL
614D 8D 49 03           STA  E
6150 AD 46 03           LDA  DELTX+1
6153 20 E8 61           JSR  COMPH
6156 8D 4A 03           STA  E+1
6159 38                 SEC            ;VER SI NEGATIVO
615A 30 01              BMI  MV4
615C 18                 CLC
615D 6E 4A 03 MV4       ROR  E+1       ;DIVIDIR POR 2
6160 6E 49 03           ROR  E
6163 A0 00              LDY  #0        ;Y=0
6165 8C 4D 03           STY  C         ;CONTADOR A CERO
6168 8C 4E 03           STY  C+1
616B F0 37              BEQ  MV7       ;BIFURCACION ABSOLUTA
                     ;
                     ;*** BUCLE PRINCIPAL DE DIBUJO ***
                     ;
616D AE 4F 03 MV5       LDX  I         ;OBTENER DIRECCION EN X
6170 18                 CLC            ;CALC. E=E+DELTY
6171 AD 49 03           LDA  E
6174 6D 47 03           ADC  DELTY
6177 8D 49 03           STA  E         ;PRIMER BYTE BAJO
617A AD 4A 03           LDA  E+1
617D 6D 48 03           ADC  DELTY+1
6180 8D 4A 03           STA  E+1
6183 30 14              BMI  MV6
6185 38                 SEC            ;CALC. E=E-DELTX
6186 AD 49 03           LDA  E
6189 ED 45 03           SBC  DELTX
618C 8D 49 03           STA  E
618F AD 4A 03           LDA  E+1
6192 ED 46 03           SBC  DELTX+1
6195 8D 4A 03           STA  E+1
6198 E8                 INX            ;X INC.
6199 20 BA 61 MV6       JSR  OUTPLT    ;SACAR UN MOVIMIENTO
619C EE 4D 03           INC  C         ;INC CONTADOR
619F D0 03              BNE  MV7
61A1 EE 4E 03           INC  C+1
                     ;
                     ;SALTAR AQUI EN EL 1ER PASO
                     ;
61A4 B1 FD    MV7       LDA  (BYT),Y ;ENCENDER UN PUNTO
61A6 0D 44 03           ORA  BITNO
61A9 91 FD              STA  (BYT),Y
61AB AD 4D 03           LDA  C         ;HECHO CUANDO C > = DELTX
61AE CD 45 03           CMP  DELTX
61B1 AD 4E 03           LDA  C+1
61B4 ED 46 03           SBC  DELTX+1
61B7 90 B4              BCC  MV5
61B9 60                 RTS            ;TERMINADO
```

```
DIR. HEXADEC. ETIQUETA COD OPER       ;COM. PRG: "ALTA RESOLUCION "
                     ;
                     ;******* OUTPLT *********
                     ;
                     ;REALIZAR UN MOVIMIENTO ELEMENTAL
                     ;
61BA 0A          OUTPLT    TXA
61BB 0A                    ASL   A          ;MULT POR 2 PARA INDICE
61BC AA                    TAX
61BD BD 0A 63              LDA   MOVTAB+1,X  ;OBTENER EL VECTOR
61C0 48                    PHA              ;BYTE ALTO EN STACK
61C1 BD 09 63              LDA   MOVTAB,X
61C4 48                    PHA              ;BYTE BAJO EN STACK
61C5 60                    RTS              ;REALIZAR EL SALTO CALCULADO
                     ;
                     ;VOLVER VIA RTS A JSR OUTPLT
                     ;
                     ;****** RNGCHK ********
                     ;
                     ;TEST DE REBOSAMIENTO DE X2,Y2
                     ;VUELTA AL PROG PRINCIPAL EN REBOSAMIENTO
61C6 AD 3E 03 RNGCHK       LDA   X2         ;TEST X2 BAJO
61C9 C9 40                 CMP   #<XMAX
61CB AD 3F 03              LDA   X2+1       ;TEST ALTO
61CE E9 01                 SBC   #>XMAX
61D0 B0 0C                 BCS   RNG2       ;X2 >XMAX, SALIR
61D2 AD 42 03 RNG1         LDA   Y2         ;TEST Y2 BAJO
61D5 C9 C8                 CMP   #<YMAX
61D7 AD 43 03              LDA   Y2+1       ;TEST BYTE ALTO
61DA E9 00                 SBC   #>YMAX
61DC 90 08                 BCC   RNG3       ;Y2 < YMAX, OK
61DE 20 BD 62 RNG2         JSR   HRREST     ;NORMAL
61E1 A0 0E                 LDY   #14        ;ILLEGAL QUANTITY ERROR
61E3 6C 00 03              JMP   (ERRVEC)   ;A RUTINA DE ERROR
61E6 60          RNG3      RTS
                     ;
                     ;********* COMPL,H ****************
                     ;
                     ;CALCULAR EL COMPLEMENTO DE UN NU-
                     ;MERO CON SIGNO.
                     ;SALTAR PRIMERO A COMPL PARA BYTE
                     ;BAJO Y LUEGO A COMPH PARA BYTE ALTO
                     ;
                     ;RESULTADO EN A
                     ;
61E7 38          COMPL     SEC              ;PARA BYTE BAJO
61E8 49 FF       COMPH     EOR   #$FF           ;COMPLEMENTA
61EA 69 00                 ADC   #0         ;SUMAR EL ESTADO DEL CARRY
61EC 60                    RTS
                     ;
61ED 20 32 62 LL           JSR   LEFT
61F0 A5 FD       DOWN      LDA   BYT
61F2 29 07                 AND   #7         ;VER 3 BITS BAJOS
61F4 49 07                 EOR   #7         ;INVERTIR
61F6 F0 08                 BEQ   DN2        ;EL BYTE ORIGINAL ERA XXXX111
61F8 E6 FD                 INC   BYT
61FA D0 11                 BNE   DN3
61FC E6 FE                 INC   BYT+1
61FE D0 0D                 BNE   DN3        ;SALTO INCONDICIONAL
6200 18          DN2       CLC              ;SUMA 320-7
6201 A5 FD                 LDA   BYT
6203 69 39                 ADC   #<313
6205 85 FD                 STA   BYT
6207 A5 FE                 LDA   BYT+1
6209 69 01                 ADC   #>313
620B 85 FE                 STA   BYT+1
620D 60          DN3       RTS
                     ;
620E 20 48 62 UR           JSR   RIGHT      ;PRIMERO DERECHA,
6211 A5 FD       UP        LDA   BYT        ;LUEGO ARRIBA
6213 29 07                 AND   #7         ;TEST BITS BAJOS
6215 D0 0F                 BNE   UP1        ;SI BYTE NO ERA XXXXX000
6217 38                    SEC              ;RESTAR 320-7
6218 A5 FD                 LDA   BYT
621A E9 39                 SBC   #<313
621C 85 FD                 STA   BYT
621E A5 FE                 LDA   BYT+1
6220 E9 01                 SBC   #>313
6222 85 FE                 STA   BYT+1
```

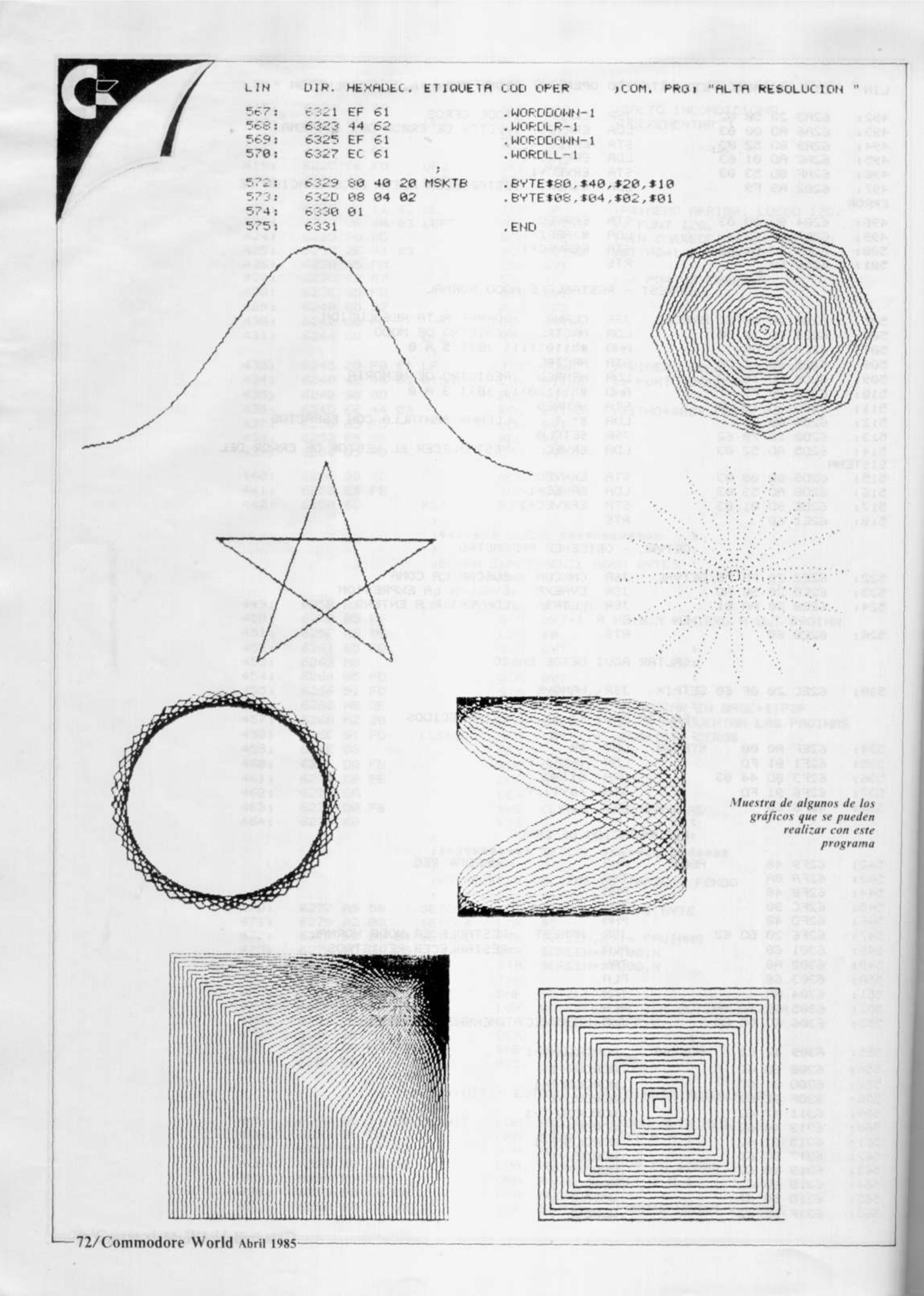


```
LIN    DIR. HEXADEC. ETIQUETA COD OPER      ;COM. PRG: "ALTA RESOLUCION "

       6224 D0 08              BNE   UP3       ;SALTO INCONDICIONAL
       6226 A5 FD     UP1      LDA   BYT       ;DECREMENTAR
       6228 D0 02              BNE   UP2
       622A C6 FE              DEC   BYT+1
       622C C6 FD     UP2      DEC   BYT
       622E 60        UP3      RTS
                      ;
       622F 20 11 62 UL        JSR   UP        ;PRIMERO ARRIBA, LUEGO IZQ.
       6232 0E 44 03 LEFT      ASL   BITNO     ;1 PUNT IZQ.
       6235 90 0D              BCC   LF2       ;SIN CORRECCION EN CARRY 0
       6237 2E 44 03           ROL   BITNO     ;BITNO=1 Y CARRY=0
       623A A5 FD              LDA   BYT
       623C E9 07              SBC   #7        ;-8 PORQUE CARRY=0
       623E 85 FD              STA   BYT
       6240 B0 02              BCS   LF2
       6242 C6 FE              DEC   BYT+1
       6244 60        LF2      RTS
                      ;
       6245 20 F0 61 LR        JSR   DOWN      ;PRIMERO ABAJO LUEGO DER
       6248 4E 44 03 RIGHT     LSR   BITNO     ;1 PUNTO DERECHA
       624B 90 0D              BCC   RGT1
       624D 6E 44 03           ROR   BITNO     ;BITNO=$80Y CARRY =0
       6250 A5 FD              LDA   BYT
       6252 69 08              ADC   #8
       6254 85 FD              STA   BYT
       6256 90 02              BCC   RGT1
       6258 E6 FE              INC   BYT+1
       625A 60        RGT1     RTS
                      ;
                      ;******* CLRHR ***********
                      ;
                      ;BORRA EXACTAMENTE 8000 BYTES
                      ;DEJA Y=X=0
                      ;
       625B A9 3F     CLRHR    LDA   #>HRLAST
       625D 85 FE              STA   BYT+1     ;INIC. PUNTERO A ULT PAGINA
       625F A9 00              LDA   #0
       6261 85 FD              STA   BYT
       6263 A8                 TAY
       6264 85 FD              STA   BYT
       6266 91 FD              STA   (BYT),Y
       6268 A0 3F              LDY   #<HRLAST ;EMPEZAR EN BASE+$1F3F
       626A A2 20              LDX   #$20      ;EN X SE CUENTAN LAS PAGINAS
       626C 91 FD     CLRHR1   STA   (BYT),Y ;ESCRIBIR CEROS
       626E 88                 DEY
       626F D0 FB              BNE   CLRHR1
       6271 C6 FE              DEC   BYT+1
       6273 CA                 DEX
       6274 D0 F6              BNE   CLRHR1    ;32 PAGINAS
       6276 60                 RTS
                      ;
                      ;*********** SETCOL **************
                      ;
                      ;ESTABLECER COLORES DE TRAZO Y FONDO
                      ;
       6277 A9 50     SETCOL   LDA   #COLOR    ;EN 1 BYTE
       6279 A2 00     SETCL0   LDX   #0
       627B 9D 00 04 SETCL1    STA   SCREEN,X ;4 PAGINAS
       627E 9D 00 05           STA   SCREEN+$0100,X
       6281 9D 00 06           STA   SCREEN+$0200,X
       6284 E8                 INX
       6285 D0 F4              BNE   SETCL1
       6287 A2 E8              LDX   #<SCREEND+1 ;LA ULTIMA PAGINA
       6289 9D FF 06 SETCL2    STA   SCREEN+$02FF,X
       628C CA                 DEX
       628D D0 FA              BNE   SETCL2
       628F 60                 RTS
                      ;
                      ;HRINIT - ESTABLECE MODO ALTA RESOLUCION
                      ;
       6290 AD 11 D0 HRINIT    LDA   HRCTRL    ;MODO ALTA RES.
       6293 09 20              ORA   #$20      ;BIT 5 A 1
       6295 8D 11 D0           STA   HRCTRL
       6298 AD 18 D0           LDA   HRMREG    ;BYT EN $2000
       629B 09 08              ORA   #$08      ;BIT 3 A 1
       629D 8D 18 D0           STA   HRMREG
       62A0 20 77 62           JSR   SETCOL    ;FORZAR NEGRO Y VERDE
```

```
LIN    DIR. HEXADEC. ETIQUETA COD OPER        ;COM. PRO: "ALTA RESOLUCION "

       62A3 20 5B 62           JSR  CLRHR   ;TODO CEROS
       62A6 AD 00 03           LDA  ERRVEC  ;VECTOR DE ERROR DEL SISTEMA
       62A9 8D 52 03           STA  ERVEC
       62AC AD 01 03           LDA  ERRVEC+1
       62AF 8D 53 03           STA  ERVEC+1
       62B2 A9 F9              LDA  #<ABRT  ;ESTABLECER NUEVA RECUPERACION DE
ERROR
       62B4 8D 00 03           STA  ERRVEC
       62B7 A9 62              LDA  #>ABRT
       62B9 8D 01 03           STA  ERRVEC+1
       62BC 60                 RTS
                      ;
                      ;HRREST - RESTABLECE MODO NORMAL
                      ;
       62BD 20 5B 62 HRREST    JSR  CLRHR   ;BORRAR ALTA RESOLUCION
       62C0 AD 11 D0           LDA  HRCTRL  ;REGISTRO DE MODO
       62C3 29 DF              AND  #%11011111 ;BIT 5 A 0
       62C5 8D 11 D0           STA  HRCTRL
       62C8 AD 18 D0           LDA  HRMREG  ;REGISTRO DE MEMORIA
       62CB 29 F7              AND  #%11110111 ;BIT 3 A 0
       62CD 8D 18 D0           STA  HRMREG
       62D0 A9 20              LDA  #" "    ;LLENAR PANTALLA CON ESPACIOS
       62D2 20 79 62           JSR  SETCL0
       62D5 AD 52 03           LDA  ERVEC   ;RESTABLECER EL VECTOR DE ERROR DEL
SISTEMA
       62D8 8D 00 03           STA  ERRVEC
       62DB AD 53 03           LDA  ERVEC+1
       62DE 8D 01 03           STA  ERRVEC+1
       62E1 60                 RTS
                      ;
                      ;GETVAL - OBTEENER PARAMETRO
                      ;
       62E2 20 FD AE GETVAL    JSR  CHKCOM  ;BUSCAR LA COMA
       62E5 20 9E AD           JSR  EVAEXP  ;EVALUAR LA EXPRESION
       62E8 20 AA B1           JSR  FLTFIX  ;CONVERTIR A ENTERO
                      ;EN Y Y EN A
       62EB 60                 RTS
                      ;
                      ;SALTAR AQUI DESDE BASIC
                      ;
       62EC 20 0F 60 SETPIX    JSR  HRMOVE
                      ;
                      ;SALTAR AQUI SI X1,Y1 ESTABLECIDOS
                      ;
       62EF A0 00     STPIX0   LDY  #0
       62F1 B1 FD              LDA  (BYT),Y
       62F3 0D 44 03           ORA  BITNO
       62F6 91 FD              STA  (BYT),Y
       62F8 60                 RTS
                      ;
                      ;RECUPERACION DE ERROR
                      ;
       62F9 48        ABRT     PHA          ;GUARDAR REG
       62FA 8A                 TXA
       62FB 48                 PHA
       62FC 98                 TYA
       62FD 48                 PHA
       62FE 20 BD 62           JSR  HRREST  ;RESTABLECER MODO NORMAL
       6301 68                 PLA          ;RESTABLECER REGISTROS
       6302 A8                 TAY
       6303 68                 PLA
       6304 AA                 TAX
       6305 68                 PLA
       6306 6C 00 03           JMP  (ERRVEC) ;MENSAJE DE ERROR
                      ;
       6309 47 62     MOVTAB   .WORDRIGHT-1
       630B 0D 62              .WORDUR-1
       630D 31 62              .WORDLEFT-1
       630F 2E 62              .WORDUL-1
       6311 47 62              .WORDRIGHT-1
       6313 44 62              .WORDLR-1
       6315 31 62              .WORDLEFT-1
       6317 EC 61              .WORDLL-1
       6319 10 62              .WORDUP-1
       631B 0D 62              .WORDUR-1
       631D 10 62              .WORDUP-1
       631F 2E 62              .WORDUL-1
```



```
DIR. HEXADEC. ETIQUETA COD OPER          ;COM. PRG: "ALTA RESOLUCION "
6321 EF 61                  .WORDDOWN-1
6323 44 62                  .WORDLR-1
6325 EF 61                  .WORDDOWN-1
6327 EC 61                  .WORDLL-1
                    ;
6329 80 40 20 MSKTB         .BYTE$80,$40,$20,$10
632D 08 04 02               .BYTE$08,$04,$02,$01
6330 01
6331                        .END
```

Muestra de algunos de los gráficos que se pueden realizar con este programa

```
PROGRAMA: "ALTA RESOLUCION "

10 I=24576
20 READ A:IF A=256 THEN END
30 POKE I,A:I=I+1:GOTO 20
24576 DATA 76,144,98,76,189,98,76,91
24584 DATA 98,76,137,96,76,236,98,32
24592 DATA 226,98,140,60,3,140,62,3
24600 DATA 141,61,3,141,63,3,32,226
24608 DATA 98,140,64,3,140,66,3,141
24616 DATA 67,3,32,198,97,169,0,133
24624 DATA 254,56,169,199,237,64,3,72
24632 DATA 41,248,10,38,254,10,38,254
24640 DATA 10,38,254,72,141,80,3,165
24648 DATA 254,141,81,3,104,10,38,254
24656 DATA 10,38,254,109,80,3,133,253
24664 DATA 165,254,109,81,3,133,254,173
24672 DATA 60,3,41,248,101,253,133,253
24680 DATA 173,61,3,101,254,133,254,104
24688 DATA 41,7,101,253,133,253,165,254
24696 DATA 105,32,133,254,173,60,3,41
24704 DATA 7,170,189,41,99,141,68,3
24712 DATA 96,32,226,98,140,62,3,141
24720 DATA 63,3,32,226,98,140,66,3
24728 DATA 141,67,3,32,198,97,56,173
24736 DATA 62,3,237,60,3,141,69,3
24744 DATA 173,63,3,237,61,3,141,70
24752 DATA 3,56,173,66,3,237,64,3
24760 DATA 141,71,3,173,67,3,237,65
24768 DATA 3,141,72,3,173,62,3,141
24776 DATA 60,3,173,63,3,141,61,3
24784 DATA 173,66,3,141,64,3,173,67
24792 DATA 3,141,65,3,169,0,141,79
24800 DATA 3,44,70,3,16,23,173,69
24808 DATA 3,32,231,97,141,69,3,173
24816 DATA 70,3,32,232,97,141,70,3
24824 DATA 169,2,141,79,3,44,72,3
24832 DATA 16,27,173,71,3,32,231,97
24840 DATA 141,71,3,173,72,3,32,232
24848 DATA 97,141,72,3,24,173,79,3
24856 DATA 105,4,141,79,3,174,69,3
24864 DATA 236,71,3,173,70,3,168,237
24872 DATA 72,3,16,27,173,71,3,141
24880 DATA 69,3,173,72,3,141,70,3
24888 DATA 142,71,3,140,72,3,24,173
24896 DATA 79,3,105,8,141,79,3,173
24904 DATA 69,3,32,231,97,141,73,3
24912 DATA 173,70,3,32,232,97,141,74
24920 DATA 3,56,48,1,24,110,74,3
24928 DATA 110,73,3,160,0,140,77,3
24936 DATA 140,78,3,240,55,174,79,3
24944 DATA 24,173,73,3,109,71,3,141
24952 DATA 73,3,173,74,3,109,72,3
24960 DATA 141,74,3,48,20,56,173,73
24968 DATA 3,237,69,3,141,73,3,173
24976 DATA 74,3,237,70,3,141,74,3
24984 DATA 232,32,186,97,238,77,3,208
24992 DATA 3,238,78,3,177,253,13,68
25000 DATA 3,145,253,173,77,3,205,69
25008 DATA 3,173,78,3,237,70,3,144
25016 DATA 180,96,138,10,170,189,10,99
25024 DATA 72,189,9,99,72,96,173,62
25032 DATA 3,201,64,173,63,3,233,1
25040 DATA 176,12,173,66,3,201,200,173
25048 DATA 67,3,233,0,144,8,32,189
25056 DATA 98,160,14,108,0,3,96,56
25064 DATA 73,255,105,0,96,32,50,98
25072 DATA 165,253,41,7,73,7,240,8
25080 DATA 230,253,208,17,230,254,208,13
25088 DATA 24,165,253,105,57,133,253,165
25096 DATA 254,105,1,133,254,96,32,72
25104 DATA 98,165,253,41,7,208,15,56
25112 DATA 165,253,233,57,133,253,165,254
25120 DATA 233,1,133,254,208,8,165,253
25128 DATA 208,2,198,254,198,253,96,32
25136 DATA 17,98,14,68,3,144,13,46
25144 DATA 68,3,165,253,233,7,133,253
25152 DATA 176,2,198,254,96,32,240,97
25160 DATA 78,68,3,144,13,110,68,3
25168 DATA 165,253,105,8,133,253,144,2
25176 DATA 230,254,96,169,63,133,254,169
25184 DATA 0,133,253,168,133,253,145,253
25192 DATA 160,63,162,32,145,253,136,208
25200 DATA 251,198,254,202,208,246,96,169
25208 DATA 80,162,0,157,0,4,157,0
25216 DATA 5,157,0,6,232,208,244,162
25224 DATA 232,157,255,6,202,208,250,96
25232 DATA 173,17,208,9,32,141,17,208
25240 DATA 173,24,208,9,8,141,24,208
25248 DATA 32,119,98,32,91,98,173,0
25256 DATA 3,141,82,3,173,1,3,141
25264 DATA 83,3,169,249,141,0,3,169
25272 DATA 98,141,1,3,96,32,91,98
25280 DATA 173,17,208,41,223,141,17,208
25288 DATA 173,24,208,41,247,141,24,208
25296 DATA 169,32,32,121,98,173,82,3
25304 DATA 141,0,3,173,83,3,141,1
25312 DATA 3,96,32,253,174,32,158,173
25320 DATA 32,170,177,96,32,15,96,160
25328 DATA 0,177,253,13,68,3,145,253
25336 DATA 96,72,138,72,152,72,32,189
25344 DATA 98,104,168,104,170,104,108,0
25352 DATA 3,71,98,13,98,49,98,46
25360 DATA 98,71,98,68,98,49,98,236
25368 DATA 97,16,98,13,98,16,98,46
25376 DATA 98,239,97,68,98,239,97,236
25384 DATA 97,128,64,32,16,8,4,2
25392 DATA 1,256
```

```
PROGRAMA: "ALTA RESOLUCION "

10 FORT=1TO2500:NEXTT
20 BA=6*16↑3:REM DIRECCION DE BASE
30 IN=BA:REM (IN) INICIALIZA AREA HI-RES
40 RS=BA+3:REM (RS) VUELVE AL MODO DE
TEXTO NORMAL
50 CL=BA+6:REM (CL) LIMPIA LA PANTALLA
HI-RES
60 DR=BA+9:REM (DR) DIBUJA UNA LINEA
ENTRE EL POINTER HI-RES Y EL PUNTO DADO
70 PX=BA+12:REM (PX) DIBUJA UN PUNTO EN
LAS COORDENADAS ESTABLECIDAS
80 MV=BA+15:REM (MV) MUEVE EL POINTER
HI-RES A OTRA POSICION SIN DIBUJAR
85 CR=BA+631:REM RUTINA PARA CAMBIAR EL
COLOR
86 POKECR+1,1:SYSCR:REM CAMBIO DE COLOR,
DIBUJO EN NEGRO SOBRE FONDO BLANCO
90 SYS(IN)
100 S=3:SYS(MV),S,S:FORI=STO195STEPS
110 X1=S:Y1=X1:X2=X1:Y2=Y1+I
120 X3=X2+I:Y3=Y2:X4=X3:Y4=Y3-I
130 SYS(DR),X2,198
140 SYS(DR),X3,Y3
150 SYS(DR),X4,Y4
160 SYS(DR),X1,Y1
170 NEXTI
180 FORT=1TO2500:NEXTT
200 R=80:XC=160:YC=100:A=π/180:S=5
210 SYS(CL)
220 FORAN=0TOπ/1.99STEPπ/20
230 SYS(MV),XC+R*SIN(AN),YC+R*SIN(AN)
240 FORI=STO360STEPS
```

(Pasa a página 78)

Por EA4ALI

Telegrafía con ordenadores

No estaría completa una serie de programas para los radioaficionados si no encluyera un programa que permitiese practicar tanto la emisión como la recepción de código morse con la ayuda de nuestros ordenadores personales.

He recibido gran cantidad de cartas a raíz de la publicación del primer artículo de esta pequeña serie, y en primer lugar debe pedir disculpas a todos por no poder contestar inmediatamente a cada uno; espero que lo comprenderéis y seréis pacientes esperando la respuesta..., uno es humano y no puede hacerlo todo a la vez.

Para que sirva de adelanto en la respuesta a muchas de las cartas, contestaré a las preguntas más repetidas:

1. Intentaré preparar los esquemas de los modems de RTTY y CW tan pronto como sea posible, pero, como muchos imaginarán, al diseñarlos, hice un esquema teórico que, tras muchas modificaciones y pruebas, en nada se parece al definitivo (supongo que los aficionados a realizar sus circuitos sabrán que esto es normal).

2. El dibujo de la placa de circuito impreso no lo tengo en limpio, ya que cumpliendo los postulados de la ley de Murphy olvidé los taladros y pistas que unían varios componentes, y a esto se unieron los 'componentes flotantes' que fueron añadidos al diseño original cuando la radiofrecuencia comenzó a confirmar los postulados de Murphy que dicen aquello de que los osciladores no arrancan y los amplificadores y filtros auto-oscilan.

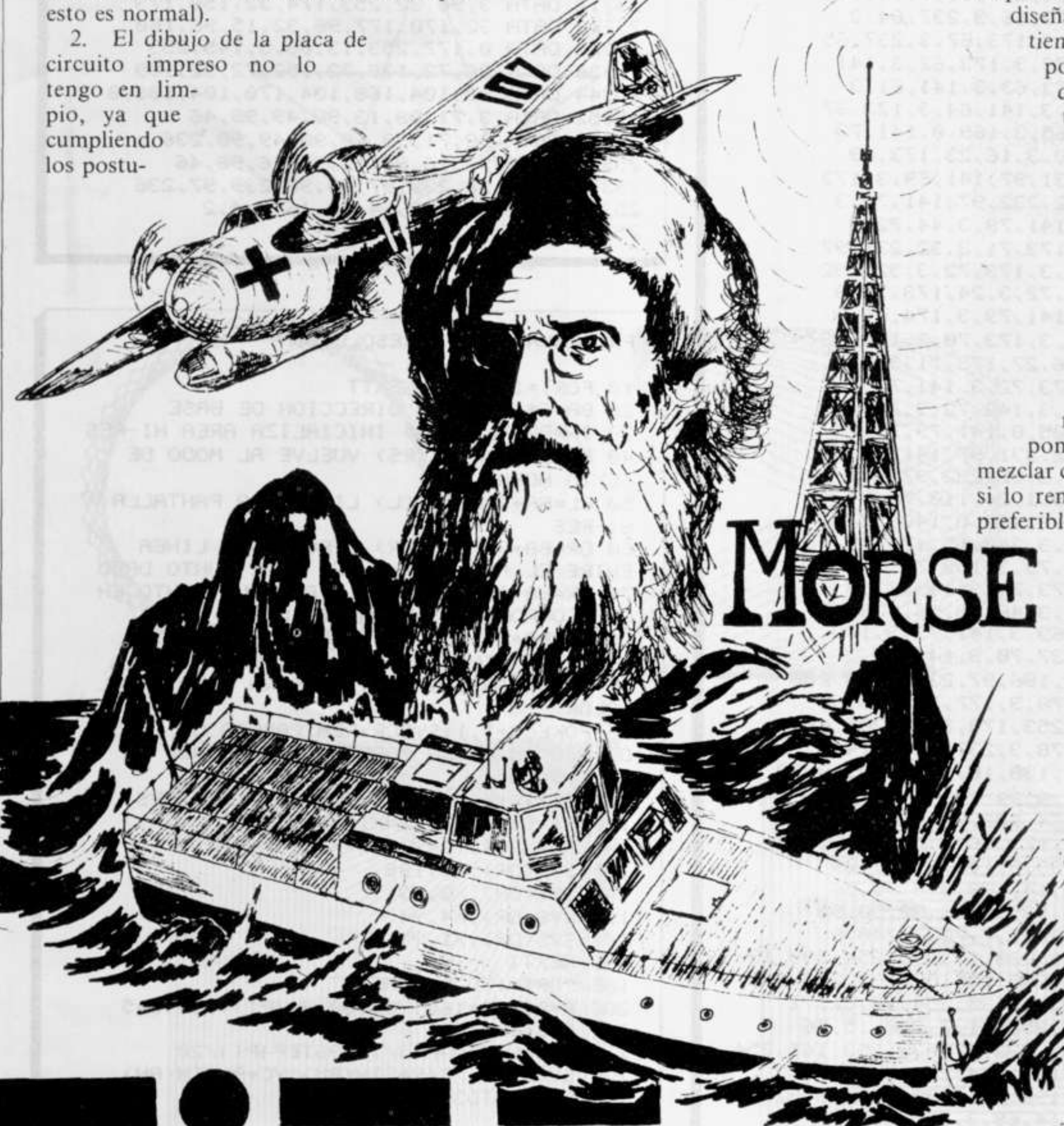
3. De los dos puntos anteriores se puede deducir que actualmente no dispongo del esquema teórico y práctico definitivo para poder publicarlo. El teórico lo podré obtener un fin de semana de éstos si desguazo mi modem y voy anotando los cambios realizados, luego sólo quedará pasarlo a un dibujo limpio y contrastado para imprenta.

Pero no ocurre lo mismo con el esquema práctico, ya que dejé de diseñar circuitos impresos hace tiempo y le perdí el tranquillo, por eso no puedo prometer que lo publicaré, pero intentaré hacerlo dentro de un par de números o tres, no metáis prisas por favor).

4. Muchos han preguntado cómo adaptar el programa de RTTY para telegrafía. Para todos estos, la respuesta está en este número, es otro programa distinto que permite emitir y recibir C.W. Quien disponga de tiempo y lo quiera mezclar con el RTTY puede hacerlo si lo renumera, pero creo que es preferible tenerlos separados ya que de este modo cargarán más rápidos de cinta o disco, además este programa altera el contenido de varios de los registros del port de usuario, y esto podría dar lugar a dificultades con el programa de RTTY.

Espero que esto sirva de contestación para la mayoría de los colegas que me han escrito; no obstante iré contestando por carta poco a poco y, para no manteneros en suspense, paso a contaros cómo funciona el programa y cómo utilizarlo.

Las líneas 1 a 10 inicializan los valores del chip de sonido y del port de usua-



rio para adaptarlo a nuestras necesidades. En los comentarios del programa podéis ver lo que hace cada una, y, si queréis, podéis cambiar la nota que se escucha en el altavoz de TV cambiando el valor de la variable NO (nota) en la línea 9.

Las líneas 11 y 12 inicializan las dos matrices con los códigos de emisión (TX$) y recepción (RX$). Los datos están contenidos en las líneas 84 y siguientes.

Las líneas 13 a 30 señalan cuáles son las teclas a las que se les han asignado códigos especiales y entran el indicativo y los mensajes asignados a las teclas de función.

La línea 31 fija la longitud de un punto en emisión (LP) según la velocidad escogida.

Las líneas 32 a 58 son la rutina de recepción de caracteres (en los REM del programa cuento lo que hace cada una), y siguen el algoritmo típico de simular dos registradores de desplazamiento de entrada serie/salida paralelo. Este es el procedimiento habitual cuando todo se hace por medio de hardware, pero es más elegante hacerlo por soft si el ordenador no tiene disponibles este tipo de entradas.

Hablando en términos técnicos de hardware, a cada impulso de reloj se desplazan los dos registradores en serie, y el dato aportado a sus entradas es el inverso uno respecto a otro, será un uno en el registro serie de los puntos si el tiempo que ha estado pulsado el manipulador es inferior al período del reloj, y será un uno en el registro serie de las rayas si el tiempo fue mayor. El período de la señal de reloj se adapta en función de la duración de los signos recibidos, de este modo permite que la velocidad en recepción se regule automáticamente en función de la señal recibida, y finalmente el carácter recibido se calcula multiplicando por dos el contenido de la salida paralelo de rayas y sumándole el de puntos.

SI NO HABEIS ENTENDIDO NADA DEL PARRAFO ANTERIOR NO DEBEIS PREOCUPAROS; el programa funciona bien aunque no sepáis cómo lo hace; los detalles van destinados a aquella minoría de élite que domina la electrónica digital y la informática y quieren saber cómo y por qué un programa funciona de tal o cual modo.

En las líneas 33, 37 y 40 he añadido unas sentencias POKE a la memoria de pantalla, lo hice para facilitar mi propia práctica con el 'machacapiñones', viendo el contenido de las variables que cuentan la duración de un punto de una raya y la media de tiempo por medio del carácter que aparece en cada una de las posiciones. Tampoco tenéis que hacerles mucho caso a esto (podéis quitarlo si queréis). Los que tengan el Kernal versión 2 (antiguo) no verán los caracteres si no hacen el correspondiente POKE a la memoria de color (en la versión 03 lo hace el propio 64).

**** Los que tengan el VIC deberán cambiar estos POKES a la posición que ocupe su pantalla (depende de RAM de que dispongan), o, si prefieren, pueden quitarlos.

Las líneas 59 a 79 son la rutina de emisión, y creo que no tiene ningún misterio.

La línea 61 comprueba si se ha pulsado una de las teclas de función (mensajes programados).

La línea 62 pasa a recepción si se pulsa la tecla R con la tecla SHIFT.

La línea 63 comprueba si el símbolo es la arroba (@) y envía el di-di-di-da-di-da (VA) pasando después a recepción.

La 64 hace una pausa de una duración equivalente a siete veces la del punto; ésta es la separación de palabras.

La 65 cambia la velocidad de emisión si se pulsa la tecla V con la tecla SHIFT.

La 67 comprueba si el carácter es un signo que esté permitido y vuelve al lazo de emisión.

La rutina de las líneas 69 a 75 es la que envía cada carácter al port de usuario haciendo al mismo tiempo que lo escuchemos en el altavoz del TV.

La rutina de la línea 76 es la que se encarga de enviar los mensajes preprogramados (el CQ CQ de... y los otros tres que seleccionemos).

La rutina de la línea 81 es la que introduce la velocidad de emisión, que debe ser de uno a cuarenta caracteres por minuto.

Las líneas DATA de la 84 a la 87 contienen los datos de la matriz de recepción (RX$), y de la 88 a la 93 los de emisión (TX$).

Si queréis asignar funciones diferentes a las teclas podéis hacerlo cambiando estas líneas.

**** MUCHO CUIDADO CON OLVIDAR NINGUNA COMA, UNO, CERO O TRES, un solo error puede hacer que la letra Z (--..) suene como G (--.) o algo similar.

Aunque el programa funcione perfectamente en el VIC-20, puede que en el C-64 dé problemas, ya que es bastante más lento (aproximadamente un 20%). Estos problemas no deben apreciarse mientras las estaciones que recibamos emitan con una cadencia regular y con los signos guardando la relación correcta (3/1), pero sí aparecerán si la manipulación del corresponsal es deficiente y además emite a más de 18 palabras por minuto (mal emitido ya que al no saber guardar una cadencia regular deberían reducir su velocidad). Este inconveniente puede solventarse haciendo que el lazo contador de recepción funcione más rápido para tener más precisión en la detección (al ser mayores las cifras alcanzadas por los contadores el error es menor). Existen varios modos de conseguir mayor velocidad; uno de ellos es pasar el programa a código máquina, pero ni yo mismo me he entretenido en hacerlo, ya que la otra solución es mucho más sencilla y cómoda... COMPILAR el programa con cualquiera de los compiladores que están disponibles en el mercado nacional (Blitz, Pet-Speed, Austro-Compiler, Simon's-Compiler, etc.). Yo he utilizado el primero de ellos y la velocidad de ejecución del lazo de recepción se ha visto incrementada unas 10 ó 20 veces; con ello permite alcanzar velocidades de recepción de unos 50 caracteres por minuto con la precisión que antes recibía a 5 ó 6 caracteres por minuto.

Tanto el VIC como el 64 no recibirán correctamente el código morse si la manipulación es defectuosa, ya que aunque el margen de tolerancia en la duración de los signos es bastante amplio, todo tiene un límite (aunque se compile el programa).

Un detalle más sobre el programa compilado: debemos alterar la constante que dividimos por la velocidad para calcular la duración de un punto, es el número 700 que aparece en las líneas 31 y 65, ya que si no lo hacemos, el programa emitirá mucho más rápido de lo que debe hacerlo (el número debéis calcularlo por el procedimiento de tanteo, compilar y probar).

El listado que aparece completo es para el C-64; para el VIC-20 sólo publicamos las líneas que deben cambiarse, ya que el resto del listado es el mismo, aunque conviene que le quitéis todos los REM al programa para aumentar su velocidad y reducir el tamaño.

Un detalle importante que se me quedaba en el tintero es que la entrada de señal del modem al ordenador se realiza por la línea del dato cero del port de usuario (D0 es la patilla C), y la salida del ordenador hacia el equipo se hace por la línea del dato 7 (D7 es la patilla L del port).

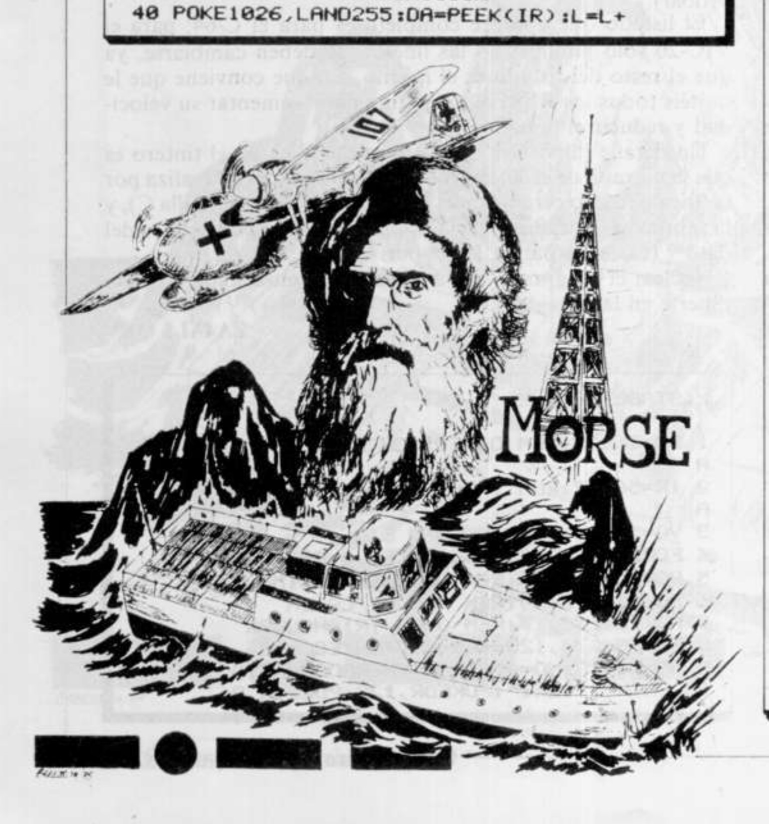
Teclear el programa y si ya tenéis algun modem para CW... ¡Suerte en la caza del DX!

EA4ALI QRV

```
LISTADO PARA EL C-64

1 DR=56579:REM DIRECTION REGISTER CI
A
2 IR=56577:REM INPUT/OUT REGISTER CI
A
3 VO=54296:REM VOLUMEN SID
4 FORZ=54272TO54296:POKEZ,0:NEXT
5 POKE54296,15:REM VOLUMEN MAXIMO
6 POKE54277,14:REM ATAQUE/CAIDA
7 POKE54276,17:REM ONDA TRIANGULAR
8 POKE54278,128:REM SOSTEN/ESCAPE
9 GE=54273:NO=40:REM GENERADOR/NOTA
10 PRINT"[CLR]":POKEDR,128:POKEIR,0:
```

```
POKE56589,63
11 DIMRX$(100),TX$(60)
12 FORZ=1TO100:READRX$(Z):NEXT:FORZ=
1TO60:READTX$(Z):NEXT
13 PRINT"[CLR][CRSRD][SPC]TECLAS[SPC]
ESPECIALES:"
14 PRINT"[SPC]#=COMPRENDIDO[SPC](SN)
"
15 PRINT"[SPC]$=ESPERA[SPC](AS)"
16 PRINT"[SPC]%=CQ"
17 PRINT"[SPC]&=FIN-MENSAJE[SPC](AR)
"
18 PRINT"[SPC]*=ERROR"
19 PRINT"[SPC]+=OK"
20 PRINT"[SPC]@=FIN-EMISION[SPC](VA)
"
21 PRINT
22 PRINT
23 PRINT
24 PRINT"PULSA[SPC]UNA[SPC]TECLA"
25 GETA$:IFA$=""THEN25
26 INPUT"[CLR][CRSRD][SPC]TU[SPC]IND
ICATIVO";ME$(0):IFME$(0)=""THEN26
27 INPUT"[CLR][CRSRD][SPC]MENSAJE[SPC]
[F3]";ME$(1):IFME$(1)=""THEN27
28 INPUT"[CLR][CRSRD][SPC]MENSAJE[SPC]
[F5]";ME$(2):IFME$(2)=""THEN28
29 INPUT"[CLR][CRSRD][SPC]MENSAJE[SPC]
[F7]";ME$(3):IFME$(3)=""THEN29
30 ME$(0)="CQ[SPC]CQ[SPC]CQ[SPC]CQ[SPC]
DE[SPC]"+ME$(0)+"[SPC]"+ME$(0)+"[SPC]
"+ME$(0)+"[SPC]PSE[SPC]K@"
31 GOSUB81:LP=INT(700/VE)
32 PRINT:PRINT"***RECEPCION***"
33 POKE1025,255AND(M/2):DA=PEEK(IR):
IFDAAND1THEN36
34 GETA$:IFA$="T"THEN59:REM PASA A T
RANSMISION
35 GOTO33:REM LAZO DE RECEPCION
36 L=0:POKEGE,NO:REM MANIPULADOR PIS
ADO
37 POKE1024,LAND255:DA=PEEK(IR):L=L+
1:IF(DAAND1)=0THEN41
38 IFL<(M/2)THEN37:REM ESPERA SEPACI
ON DE SIGNOS O RAYA
39 GETA$:IFA$="T"OR A$="[SHIFTT]"THE
N59:REM PASA A TRANSMISION
40 POKE1026,LAND255:DA=PEEK(IR):L=L+
```




```
1:IFDAAND1THEN39:REM ESPERA MANIP.SU
ELTO
41 POKEGE,0:RA=2*RA:PU=2*PU:IFL<(M/2
)THENPU=PU+1:M=1+(5*M+2*L)/6:GOTO43
42 IFL>=(M/2)THENRA=RA+1:M=1+(4*M+2*
L)/6
43 L=0
44 DA=PEEK(IR):L=L+1:IFDAAND1THEN36
45 IFL<MTHEN44
46 N=2*RA+PU
47 IFN=96THENN=90
48 IFN=105THENN=86
49 IFN=108THENN=87
50 IFN=114THENN=85
51 IFN=119THENN=88
52 IFN=255THENN=92
53 IFN>99THENN=100
54 IFN=68THEN59:REM PASA A EMISION
55 PRINTRX$(N);:PU=0:RA=0
56 DA=PEEK(IR):L=L+1:IFDAAND1THEN36
57 IFL<1.2*MTHEN56
58 PRINT"[SPC]";:GOTO33
59 POKEGE,0:PRINT:PRINT"***EMISION**
*"
60 GETA$:IFA$=""THEN60
61 O=ASC(A$):IF(O>132)AND(O<137)THEN
76:REM TECLAS DE FUNCION
62 IFO=210THEN32:REM PASA A RECEPCIO
N
63 IFO=64THENGOSUB69:GOTO32:REM PASA
 A RECEPCION
64 IFO=32THENPRINT"[SPC]";:FORTT=1TO
7*LP:NEXT:GOTO60
65 IFO=214THENGOSUB80:LP=INT(700/VE)
:GOTO60
66 IFO=13THENPRINT:GOTO60
67 IF(O<32)OR(O>90)THEN60
68 GOSUB69:GOTO60:REM LAZO DE TRANSM
ISION
69 O=O-32:P$=TX$(O):FORZ=1TOLEN(P$)
70 Q$=MID$(P$,Z,1):Q=VAL(Q$)
71 IFQ=0THENFORTT=1TOLP:NEXT:GOTO75
72 POKEIR,128:POKEGE,NO
73 FORTT=1TOLP*Q:NEXT
74 POKEIR,0:POKEGE,0:FORTT=1TOLP:NEX
T
75 NEXT:PRINTCHR$(O+32);:FORTT=1TO2*
LP:NEXT:RETURN
76 REM *** EMISION DE MENSAJES ***
77 R=O-133:FORP=1TOLEN(ME$(R))
78 Z$=MID$(ME$(R),P,1):O=ASC(Z$):IFO
=64THEN32
79 GOSUB69:NEXT:GOTO60
80 PRINT:PRINT"[22SPC]";VE
81 INPUT"[CRSRU]VELOCIDAD[SPC](DE[SPC]
1[SPC]A[SPC]40)";VE
82 IFVE<1ORVE>40THENPRINT"[CRSRU][23SPC]
20":GOTO81
83 RETURN
84 DATAE,T,I,A,N,M,S,U,R,W,D,K,G,O,H
,V,F,,L,,P,J,B,X,C,Y,Z,Q,,,5,4
85 DATACOMPRENDIDO,3,,,,2,ESPERA,,FI
N DE MENSAJE,,,,,1,6,=,/,
86 DATA,,(,,7,,,,8,,9,0,.,,,,,,FIN DE
 EMISION,,,,,,
87 DATA?,,,,,,',,,,.,",",;,),":",,-,,
ERROR,',"[SPC]",,,,,"[SPC]","[SPC]"
88 DATA0,0,11131,13111,313103313,131
31,133331,0,0,11111111
89 DATA3330313,331133,311113,131313,
31131,33333,13333,11333,11133,11113
90 DATA11111,31111,33111,33311,33331
,333111,313131,0,31113,0
91 DATA113311,111313,13,3111,3131,31
1,1
92 DATA1131,331,1111,11,1333,313,131
1,33,31,333,1331,3313,131,111,3,113
93 DATA1113,133,3113,3133,3311,0,0
```

BOXER 12

high resolution monochrome monitor 12"

NEW 85
NOVEDAD 85

ELECTRICAL ENVIRONMENTAL CHARACTERISTICS

CRT	SIZE	12"
	DEFL. ANGLE	90°
DISPLAY FORMAT	CHARACTERS	2000 (80 × 25)
VIDEO	INPUT SIGNAL	COMPOSITE VIDEO
	VIDEO SIGNAL	1 Vpp pos.
	RISE/FALL TIME	≤ 30 ns
	BANDWIDTH	20 MHz
	CENTRE RESOLUTION LINES/IN	1000
	INPUT RESISTANCE	75 Ohm
BLANKING TIME	HORIZONTAL	≤ 8 μs
	VERTICAL	≤ 700 μs
COMP. SYNC.	H. SYNC.	15.650-15.750 KHz
	V. SYNC.	50-60 Hz
EHT	(Ib = 0)	13 KV
POWER SUPPLY	INPUT VOLTAGE	min. 180 max. 264 Vac
	CONSUMPTION	30 VA
GEOMETRY	RASTER DISTORTION	max 1 %
	SCAN LINEARITY	max 10 %
	FOCUS	internal control
	V. AMPLITUDE	internal control
	V. FREQUENCY	internal control
	V. UPPER AND LOWER LINEARITY	internal control
	H. AMPLITUDE	internal control
	H. FREQUENCY	internal control
	H. LINEARITY	internal control
	H. PHASE	internal control
ENVIROMENTAL	AMBIENT TEMPERATURE	0° C + 40° C
	AMBIENT HUMIDITY (not condensed)	5-90 %
	STORAGE TEMPERATURE	40° C + 65° C
	STORAGE HUMIDITY (not condensed)	5-90 %
WEIGHT	GROSS/NET	5,7/6,6 Kg.

• *audio optional*

HANTAREX



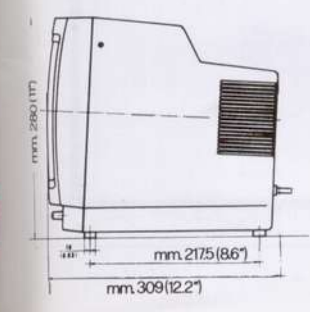
BOXER 12

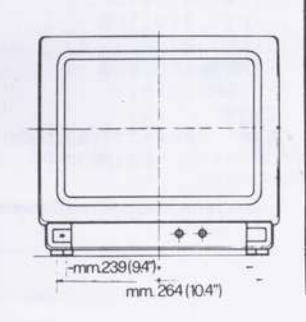


TRATTAMENTO SCHERMO: SCURO - ANTIRIFLETTENTE
SCREEN TREATMENT: DARK GLASS - ETCHED

FOSFORO - P31 - VERDE MEDIO-BREVE
PHOSPHOR - P31 - GREEN MEDIUM-SHORT

DATI MECCANICI
MECHANICAL DATA







HANTAREX®

Electronic Equipment Manufacturer

QUALITY . RELIABILITY . SERVICE

Aragón, 210, 1°, 1ª - Barcelona 11 - teléf. (93) 3232941 - telex 98017

```
PARA EL VIC-20
BORRAR LAS LINEAS 1 A 10
Y SUSTITUIRLAS POR:


1 DR=37138
2 IR=37136
5 POKE36878,15:REM VOLUMEN MAXIMO
6 POKE36874,0
9 GE=36874:NO=245:REM GENERADOR/NOTA

10 PRINT"[CLR]":POKEDR,128:POKEIR,0:
POKE37148,124
33 DA=PEEK(IR):IFDAAND1THEN36
37 DA=PEEK(IR):L=L+1:IF(DAAND1)=0THE
N41
40 DA=PEEK(IR):L=L+1:IFDAAND1THEN39:
REM ESPERA MANIP.SUELTO
```

SUMA DE CONTROL DEL PROGRAMA MORSE-64

1	84	2	109	3	112
4	130	5	45	6	92
7	87	8	13	9	17
10	221	11	233	12	227
13	149	14	163	15	35
16	243	17	113	18	238
19	255	20	160	21	153
22	153	23	153	24	239
25	52	26	179	27	87
28	92	29	97	30	6
31	99	32	68	33	172
34	117	35	111	36	89
37	8	38	139	39	116
40	38	41	182	42	90
43	46	44	126	45	230
46	192	47	10	48	54
49	58	50	53	51	61
52	57	53	52	54	177
55	177	56	126	57	38
58	97	59	121	60	51
61	90	62	82	63	232
64	30	65	85	66	89
67	186	68	100	69	84
70	97	71	156	72	31
73	5	74	137	75	238
76	30	77	148	78	137
79	225	80	70	81	235
82	237	83	142	84	31
85	68	86	95	87	191
88	46	89	211	90	157
91	151	92	166	93	162

SUMA DE CONTROL DE LOS CAMBIOS DEL VIC-20

1	78	2	81	5	51
6	255	9	79	10	1
33	31	37	173	40	201

(Viene de pág. 73)

```
250
SYS(DR),XC+R*SIN(2*I*A+AN),YC+R*SIN(I*A+
AN)
260 NEXTI,AN
270 FORT=1TO2500:NEXTT
300 SYS(CL)
310 D=4:E=2:X=XC:Y=YC
320 SYS(MV),X,Y
330 FORI=0TO20
340 D=D+E:Y=Y+D:SYS(DR),X,Y
350 D=D+E:X=X+D:SYS(DR),X,Y
360 D=D+E:Y=Y-D:SYS(DR),X,Y
370 D=D+E:X=X-D:SYS(DR),X,Y
380 NEXTI
390 FORT=1TO2500:NEXTT
400 SYS(CL):S=+PI I/3
410 FORT=0TOSSTEPS/8
420 SYS(MV),XC+R*COS(T),YC+R*SIN(T)
430 FORI=STO2*+PI ISTEPS
440 SYS(DR),XC+R*COS(I+T),YC+R*SIN(I+T)
450 NEXTI,T
460 FORT=1TO2500:NEXTT
500 SYS(CL):S=+PI I/4:D=R/20
510 FORT=0TOSSTEPS/20
520 SYS(MV),XC+R*COS(T),YC+R*SIN(T)
530 FORI=STO2*+PI ISTEPS
540 SYS(DR),XC+R*COS(I+T),YC+R*SIN(I+T)
```

```
PROGRAMA: "ALTA RESOLUCION "

550 NEXTI
560 R=R-D:NEXTT
580 FORT=1TO2500:NEXTT
600 SYS(CL):R=80:S=+PI I/8:D=R/20
610 FORT=0TOSSTEPS/40
620 SYS(PX),XC+R*COS(T),YC+R*SIN(T)
630 FORI=STO2*+PI ISTEPS
640 SYS(PX),XC+R*COS(I+T),YC+R*SIN(I+T)
```

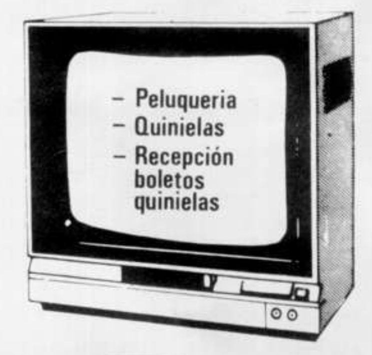
```
650 NEXTI
660 R=R-D:NEXTT
680 FORT=1TO2500:NEXTT
700 SYS(CL):R=80:S=2*+PI I/5:A=+PI I/10
710 FORI=0TO4
720 T=A+I*S
730 X(I)=XC+R*COS(T):Y(I)=YC+R*SIN(T)
740 NEXTI
750 SYS(MV),X(0),Y(0)
760 SYS(DR),X(2),Y(2):SYS(DR),X(4),Y(4)
770 SYS(DR),X(1),Y(1):SYS(DR),X(3),Y(3)
780 SYS(DR),X(0),Y(0)
790 FORT=1TO2500:NEXTT
800
SYS(CL):A=160:B=A/2:SYS(MV),0,A*EXP(-4)
810 FORX=4TO2*A-1STEP4
820 SYS(DR),X,A*EXP(-((X-A)/B)↑2)
830 NEXTX
880 FORT=1TO2500:NEXTT
890 SYSCL
900 FORX=0TO319STEP2
910 SYS(MV),160,100
920 SYS(DR),X,0
930 SYS(MV),160,100
940 SYS(DR),X,199
950 NEXTX
960 FORY=0TO199STEP2
970 SYS(MV),160,100
980 SYS(DR),0,Y
990 SYS(MV),160,100
1000 SYS(DR),319,Y
1010 NEXTY
1020 GETA$:IFA$=""THEN1020
9998 FORT=1TO2500:NEXTT
9999 SYS(RS)
```

algunos de nuestros programas para los ordenadores personales

commodore 64







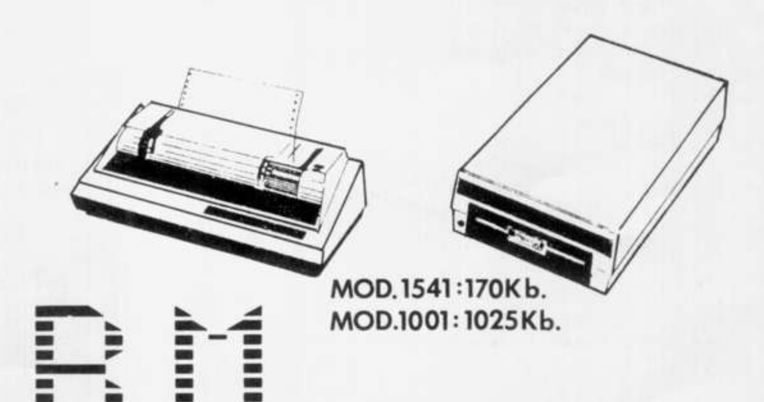






¡Aumente la capacidad de su commodore 64 hasta 1.025 Kb.





MOD.1541: 170Kb.
MOD.1001: 1025Kb.

B.M.

BASIC MICRO-ORDENADORES, S.A.

AVD. CESAR AUGUSTO, 72
Tlfs. 23 56 82 y 22 65 44
50003 ZARAGOZA

SE NECESITAN DISTRIBUIDORES

se aceptan





ESTE ES EL SIMBOLO DE COMMODORE, COMPAÑIA AMERICANA, LIDER MUNDIAL EN NUMERO DE ORDENADORES INSTALADOS.

Su Commodore 64 tiene mucho que decirle.

Unidad de Disco.

El Commodore 64 es el resultado de la experiencia internacional de Commodore como líder indiscutible en el mercado de los microordenadores.

El Commodore 64 es el ordenador más completo y potente de su categoría,... pero todavía tiene mucho que decirle.

Por ejemplo su Unidad de Disco.

Sienta como aumenta notablemente la capacidad de memoria de su C-64, como agiliza la carga y descarga de programas y facilita la localización, casi instantánea, de cualquier dato.

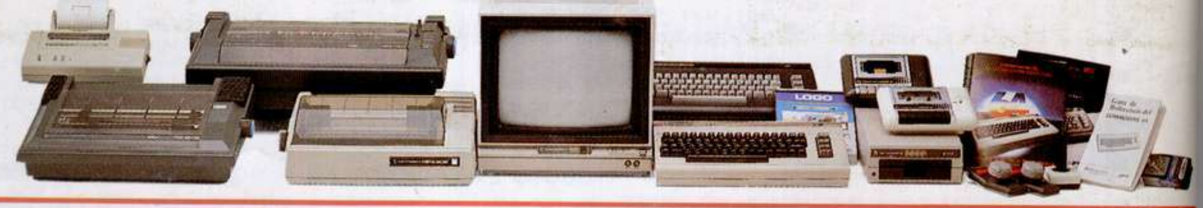
Amplíe las posibilidades de su C-64, descubriendo su extensa gama de periféricos.

Ahora que ya sabe que su Commodore 64 tiene todavía mucho que decirle, prepárese a conocerle mejor.

PRINCIPALES CARACTERISTICAS

– 170 K de capacidad – Ficheros secuenciales y relativos y de acceso directo – Unidad inteligente, con sistema operativo incorporada.





Microelectrónica y Control *c/ Valencia, 49-53 08015 Barcelona - c/ Princesa, 47 3.º G 28008 Madrid*
Unico representante de Commodore en España.